NON-PARAMETRIC AND SEMI-PARAMETRIC METHODS FOR PARSIMONIOUS STATISTICAL LEARNING WITH COMPLEX DATA

Sayan Dasgupta

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Biostatistics in the Gillings School of Global Public Health.

Chapel Hill
2014

Approved by:

Michael R. Kosorok

Yair Goldberg

Denise Esserman

Donglin Zeng

Kinh N. Truong

Stephen Cole

ABSTRACT

Sayan Dasgupta: Non-parametric and semi-parametric methods for parsimonious statistical learning with complex data
(Under the direction of Michael R. Kosorok)

In clinical research, non-parametric and semi-parametric methods are increasingly gathering importance as statistical tools to infer on accumulated data. They require fewer assumptions and their applicability is much wider than the corresponding parametric methods. Being robust, these methods are seen by some statisticians as leaving less room for improper use and misunderstanding. In this dissertation we study some of these nonparametric and semiparametric methods in statistical learning and their applications to various areas of biomedical research.

In the first part of our dissertation, we study the application of temporal process regression in the study of medical adherence. Adherence refers to the act of conforming to the recommendations made by the provider with respect to timing, dosage, and frequency of medication taking. Here we assess the effect of drug adherence in the study of viral resistance to antiviral therapy for chronic Hepatitis C. We use Temporal Process Regression (Fine, Yan, and Kosorok 2004) to model adherence as a longitudinal predictor of SVR. We show that adherence has a significant effect on SVR and this analysis can serve as an archetype for more statistically efficient analyses of medical adherence in studies where the common theme till now has been to report summary statistics.

In the second part of the dissertation, we develop an approach for feature elimination in support vector machines, based on recursive elimination of features. We present

theoretical properties of this method and show that this is uniformly consistent in finding the correct feature space under certain generalized assumptions. We present case studies to show that the assumptions are met in most practical situations and give simulation studies to demonstrate performance of the proposed approach.

In the third part of the dissertation we focus our attention to feature selection in Q-learning. Here we discuss three different methods for feature selection, based on the same vital idea of feature screening through ranking in a sequential backward selection scheme. We discussed the applicability of the methods, reasoned on heuristics stemming from our previous work on feature selection in support vector machines and gave results showing their performance in various simulated settings.

ACKNOWLEDGEMENTS

This thesis was conceived, nurtured and brought into existence from the womb of a union of multiverses with varying degrees of philosophical, social and economic influences.

First and foremost, I must thank my advisor, Dr. Michael Kosorok for guiding and helping me through my dissertation. He allowed me the space to reason and bicker with myself, and that was the motivation I badly needed. As graduate researchers, periodically we reach regions of extreme calm, or are amidst chaos, or often lost in the inflection points. Often it was my good friend Dr. Yair Goldberg, who pointed and guided me out of troubled waters, and for that and for plenty others I am truly grateful to him. I must also thank Dr. Denise Esserman for implicitly trusting in my abilities, Dr. Donglin Zeng for being technically critical, Dr. Young Troung for his heuristic ideas and Dr. Stephen Cole for his practical viewpoints.

I am also grateful to my many friends here in Chapel Hill and beyond, whose collective and individual influences shaped my life here for the last five years, and made me intrinsically feel at ease. It remains essential and also completely unnecessary to mention BA here. They encompass the single largest socio-cultural domain of my existence, which is inherent enough to deem it an insufficient prospect to express it in so few words.

I must thank my parents for their unconditional love and support, and for being the immovable pillar of encouragement from close and afar. I also thank the rest of my family for being my backbone, and an almost intrinsic safety net to fall back to even

in my darkest of times.

Finally I dedicate this thesis to my parents for giving me shape, structure, and an entity in this world, and to Ankita Roy for revealing to me the path to enlightenment.

TABLE OF CONTENTS

LIST OF TABLES

Table

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Supervised learning deals with the task of inferring a function from labeled training data. When the training data contains the subgroup information and we want to predict the future subgroups, it is a classification problem. And in cases where the training data contains the continuous response values and our aim is to predict future responses, it is a regression problem. In this dissertation we study properties of three of these supervised learning methods and their applicability in clinical studies in depth.

1.1 Temporal Regression and Medical Adherence

Adherence to, or compliance with a medication regimen, is defined as the extent to which patients take medications as prescribed by their health care providers (Osterberg and Blaschke 2005). In recent years, adherence has become a serious area of research in medicine. In this part of the dissertation we use temporal processes to study adherence and its relationship with the medical end-point in the VIRAHEP-C study. Typically information on medical adherence is gathered over time and most of the previous research on this topic has failed to incorporate this longitudinal component of adherence in their analysis. This dissertation aims to rectify this and provide an interesting insight into efficient handling of adherence data.

The data for this analysis was obtained from the NIDDK-funded VIRAHEP-C study, which enrolled 401 adults with chronic hepatitis C and genotype 1 infection at eight U.S. medical centers (Conjeevaram et al. 2006). All participants were on the combination therapy of Peginterferon and Ribavirin for up to 48 weeks. One hundred

and forty-seven of them, who showed detectable viremia at week 24, were discontinued from the therapy, while the remaining 254 participants with undetectable or indeterminate HCV RNA by 24 weeks, continued for a total of 48 weeks. Patients attended a baseline visit and then follow-ups at treatment weeks $2, 4, 8, 12$ and then monthly up to 48 weeks. In this analysis, however, we only concentrate on the first 24 week window. The endpoint of focus is Sustained Virologic Response (SVR) measured six months post treatment, defined as undetectable viremia (HCV RNA < 50 IU/mL). Details of the VIRAHEP-C protocol can be found at `https://www.niddkrepository.org/niddk/jsp/public/dataset.jsp#VIRAHEP-C`. Baseline data included socio-demographic variables (e.g., age, gender, race, marital status, education level, employment status, health insurance status, etc.) and medical variables (e.g., fibrosis level, alcohol consumption, presence of baseline antidepressant use, etc.). The Center for Epidemiologic Studies-Depression (CES-D) (Radloff (1977)) scale was used to measure depression symptoms and a visual analog scale was used to measure symptoms including (i) fatigue, (ii) headache, (iii) muscle aches and pains, (iv) irritability, (v) depression, and an (vi) overall symptom score.

Adherence (daily adherence to Ribavirin and weekly adherence to Peginterferon) was measured by the Medication Event Monitoring System (MEMS, AARDEX, Sion, Switzerland) (for details regarding the MEMs system see Liu et al. (2001)) as a weekly count for Peginterferon and a daily count for Ribavirin. If individual i took the Peginterferon shot at week j, they were considered adherent ($X_{ij} = 1$) and non adherent otherwise ($X_{ij} = 0$). Similarly if individual i took both counts of the prescribed doses of Ribavirin on day j, she/he was considered fully adherent ($\tilde{X}_{ij} = 2$). If she/he took only one count of the prescribed doses on day j, she/he would be considered partly adherent ($\tilde{X}_{ij} = 1$) and otherwise he will be considered to be nonadherent for that day ($\tilde{X}_{ij} = 0$). If a participant was prescribed to refrain from dosing for either Peginterferon

or Ribavirin and did not open the MEMS cap, she/he was considered fully adherent (See Evon et al. (2013) for details).

Thus one important goal of this analysis is to apply a method that can be generalized and implemented in situations similar to this, and that it provides an efficient and informative approach to examine adherence data. For this, we propose to use temporal processes to study adherence. The term 'temporal process' refers to a functional process that is completely specified over time. The idea of extending the marginal mean model to incorporate regression for response and covariates that are temporal processes observed over compact intervals was developed by Fine et al. (2004). It was originally intended as a robust substitute for intensity models in time-to-event data, since only the mean instead of the full stochastic structure of the processes needs to be specified. However temporal process regression is a useful formulation in longitudinal studies as well, where the response as well as covariates are observed multiple times over an interval. Conceptually, the modeling strategy is functional data analysis (Ramsay and Dalzell 1991) and is closely related to varying-coefficient models (Hastie and Tibshirani 1993) for longitudinal data at finite irregularly spaced times. The cross-sectional data at each time point is used to formulate an estimating equation in a typical linear model set-up, and the time-varying coefficients at that time point are estimated by solving it. Temporal processes have been used before in analyses in medicine. Yan et al. (2010) applied temporal process regressions to analyze progressive symptoms in a case study of the Cystic Fibrosis Foundation Patient Registry data, as an alternative to the commonly employed proportional hazards models. However, as mentioned, the set up for this analysis was right censored data, and to our knowledge, temporal processes have not been employed in linear models involving longitudinal data before. Hence this dissertation shows the importance of temporal processes in such a set up.

In analyses involving adherence data, we typically encounter data that are longitudinal in nature. For example in the VIRAHEP C study, 401 patients were followed for 24 weeks, and adherence for Peginterferon was recorded once each week, while that for Ribavirin was recorded each day. As observed above, studies on adherence, whether looking at the importance of adherence on medical end-points or analyzing factors that affect adherence in general, mostly involve sample summaries of these longitudinal data. But the drawback of this type of cross-sectional approach is multifold. First of all, it suffers from an immense loss of information, affected by compiling summary statistics pooled over the entire length of the study. Hence hypothesis tests typically proposed to objectify the causal relationships in such analyses are far less powerful. Second of all, by incorporating the temporal nature of adherence, we can observe the covariate effects across the study period which can provide further insight into the dynamic nature of this relationship across time.

The main contribution of this dissertation is providing an insightful approach to analyzing adherence data. In this dissertation, we study the effect of adherence on sustained virologic response, the end point in the VIRAHEP C study, using temporal process regression. Hence in this case, adherence is incorporated as a time-varying covariate in the regression set up and SVR is incorporated as the response and remains constant over time. It is worthwhile to note that a similar approach can be used in reverse studies where adherence is analyzed as the response while looking for meaningful factors contributing to varying trends of adherence over time. Another novel contribution of this dissertation is the approach used to create the confidence bands for the processes. In Fine et al. (2004), the authors employ bootstrapping to simulate from the empirical distribution of $\sqrt{n}(\hat{\beta}(t) - \beta_0(t))$ (the centered covariate effects) to create confidence bands for $\beta_0(t)$, the true parametric process. In this dissertation we modify this approach by utilizing the empirical distribution of $\sup_{t\in[l,u]} \sqrt{n}|\hat{\beta}(t) - \beta_0(t)|$

to bootstrap from. This method actually helps in establishing a direct relationship between these confidence bands and our proposed hypothesis tests.

1.2 Feature Elimination in Support Vector Machines

In recent years it has become increasingly easy to collect large amounts of information, especially with respect to the number of explanatory variables or 'features'. However the additional information provided by each of these features may not be significant for explaining the phenomenon at hand. Learning the functional connection between the explanatory variables and the response from such high-dimensional data can itself be quite challenging. Moreover some of these explanatory variables or features may contain redundant or noisy information and this may hamper the quality of learning. One way to overcome this problem is to use variable selection or feature elimination techniques to find a smaller set of variables or features that is able to perform the learning task sufficiently well.

In this work we discuss feature elimination in support vector machines. The popularity of support vector machines (SVM) as a set of supervised learning algorithms is motivated by the fact that SVM learning methods are easy-to-compute techniques that enable estimation under weak or no assumptions on the distribution (see Steinwart and Chirstmann 2008). SVM learning methods, which we review in detail in Section 3.1, are a collection of algorithms that attempt to minimize a regularized version of the empirical risk over some reproducing kernel Hilbert space (RKHS) with respect to some loss function. The standard SVM decision function typically utilizes all the input variables. Hence, when the input dimension is large, it can suffer from the so-called 'Curse of Dimensionality' (Hastie et al. 2001). A procedure for variable selection is thus of importance to obtain a more intelligible solution with improved efficiency. The advantages of variable selection are multi fold: it increases the generalized performance of

the learning, it clarifies the causal relationship in the input-output space, and results in reduced cost of data collection and storage and better computational properties.

One of the earliest works on variable selection in SVM was formulated by Guyon et al. (2002). They developed a backward elimination procedure based on recursive computation of the SVM learning function, known widely as recursive feature elimination (RFE). The RFE algorithm performs a recursive ranking of a given set of features. At each recursive step of the algorithm, it calculates the change in the RKHS norm of the estimated SVM function after deletion of each of the features remaining in the model, and removes the one with the lowest change in such norm, thus performing an implicit ranking of features. A number of modified approaches have been developed since then, inspired by RFE (see Rakotomamonjy 2003, Aksu et al. 2010, Aksu 2012). Alternate *wrapper*-based selection methods have also been formulated like in Weston et al. (2001), Chapelle et al. (2002). *Filters* have been used for feature elimination in SVMs in many previous works (see for example Mladenic et al. 2004, Peng et al. 2005). *Embedded methods* for variable selection include redefining the SVM training to include sparsity (Weston et al. 2003, Chan et al. 2007). For example, Bradley and Mangasarian (1998) suggested the use of the l_1 penalty to encourage feature sparsity. Zhu et al. (2003) suggested an algorithm to compute the solution path for this l_1-norm SVM efficiently. Other methods include introducing different penalty functions like the SCAD penalty (Zhang et al. 2006), the l_q penalty (Liu et al. 2007), a combination of l_0 and l_1 penalty (Liu and Wu 2007), the elastic net (Wang et al. 2006), the f_∞ norm (Zou and Yuan 2006), and using a penalty functional in the framework of the smoothing spline ANOVA (Zhang 2006).

Some of the common drawbacks of these methods include, (i) they might lack versatility in application, or (ii) might lack concrete theoretical justifications. Like, most of *embedded methods* work only in linear SVMs, that is, only when we consider a linear

functional class for the optimization. Hence the main drawback for these penalized methods (with penalty functions like l_1, l_q, elastic net, etc.) is that they can only work in linear kernels, as these become ineffectual concepts in the framework of more complex function classes like RKHSs with non linear kernels. RFE and RFE derived methods however help to address this issue, as these methods can work in complex problems as well, where we might need a larger class of functions (than just the linear space) for the optimization. Another key feature of RFE is that it does feature selection, that is, when a feature is removed, all its effects (main effects and interactions with other features) are removed. However, the most important drawback for these methods is that arguments for them have mostly been heuristic, and their ability to produce successful data-driven performances have been examined only in simulated or observed data. Hence, the theoretical properties of them have never been studied in rigorous detail. A key reason behind this lack of theory is the absence of a well-established framework for building, justifying, and collating the theoretical foundation of such a feature elimination method. This part of the dissertation aims at building such a framework and modifying RFE to create a recursive technique that can be validated as a theoretically sound procedure for feature elimination in SVMs.

Developing a theoretical structure that validates recursive feature elimination in non-linear SVMs is challenging. At each stage of the feature elimination process, we move down to a 'lower dimensional' feature space and the functional spaces need to be adjusted to cater to the appropriate version of the problem in these subspaces. Euclidean spaces, for example, as well as many specialized functional classes admit a nested structure in this regard, but that is not true in general. The SVM algorihtm attempts to minimize the empirical regularized risk within an RKHS of functions. Starting with a given RKHS, one daunting task is then to redefine the functional space on the lower dimensional domain so that it retains the reproducing property and that

these spaces remain cognate to one another. The basis for the theory on RFE depends heavily on specifying these pseudo-subspaces, and a contribution of this part of the dissertation is to formulate a way to do this.

Another contribution of this part of the dissertation is a modification of the criterion for deletion and ranking of features in Guyon et al.'s RFE to enable theoretical consistency. Here we develop a ranking of the features based on the lowest difference observed in the regularized empirical risk after removing each feature from the existing model. It is important to understand that removing a feature from the functional model, means not only that the main effect of the feature are removed, but also all complex interactions the feature might have had with the remaining ones in the model, are eliminated as well. The heuristic reasoning behind this is that if any of the features do not contribute to the model at all, the increase in the regularized risk will be inconsequential. This allows RFE to be generalized to the much broader yet simpler setting of empirical risk minimization where we can apply the same idea to empirical risk. This can thus serve as a useful starting point for the analysis of feature elimination in SVM (details are given in Appendix B.1).

In this part of the dissertation, we show that if the functional space is either nested or dense, then assumption of a null model is enough to guarantee that modified RFE is asymptotically consistent in finding the 'correct' feature space, under reasonable regularity conditions. The notion of consistency in such a context has not been defined previously, and this work aims at positing a basis for which such results are meaningful. We also show through interesting examples that in risk minimization settings for any general functional space, existence of a null model may not be enough to guarantee consistency of an algorithm based on a recursive search, and certain additional restrictions are required that may not hold in generality. The notations and the oracle bounds used in this work will closely follow the ones used and derived in the text Steinwart

and Chirstmann (2008) (hereafter abbreviated SC08).

The main body of the work is given in Section 3. We present the proposed version of RFE for SVMs, and discuss the concept of feature elimination in this framework. We discuss the assumptions required to establish consistency of RFE in the simpler yet practical settings of nested or dense models. The main consistency results are provided following that, and the implications of these results in some known settings of risk minimization in nested or dense models are discussed in depth, including the setting of risk minimization in linear models and SVM for classification with a Gaussian RBF kernel. Next, we relax the earlier assumptions to allow us to establish consistency of the algorithm in more complex functional spaces. Then we study two more interesting applications of kernel machines in imaging and protein classification and discuss how our method can be useful for feature selection there. Finally, we prove our main result under this most general setting. We provide some simulation results to demonstrate how RFE works and how it can be used in intelligent selection of features, and compare it with penalized methods for feature selection. A general discussion is provided followed by detailed proofs for important results.

1.3 Feature Elimination in Q Learning

Personalized medicine can be defined as the medical model that can adapt itself to appropriate needs of a patient, with treatments and medical decisions suited to his/her requirements. The traditional 'one size fits all' approach to treatment have been replaced with a more adaptive or 'personalized' approach. The best clinical regimes are adaptive to patients over time and tailored to the specific requirements of the individual patient. Treatment individualization and adaptation over time is also crucial for management of chronic diseases and conditions. In many cases the one treatment for the entire population strategy is not only suboptimal, but also unrealistic (see

Zhao et al. 2014). An ideal treatment rule should be adaptive, robust and tailored to the requirement of a given individual based on his/her prognosis information. With steady advance in treatment methods and a better understanding of human genetics, researchers have been able to incorporate this new information into clinical diagnosis. Research projects into human genetics have paved way for better understanding of genes in an individual's physiology and development. Researchers have now been able to discern the role of single nucleotide polymorphisms (SNPs), that account for genetic variabilities between individuals and as a result genome-wide association studies (GWAS) have been conducted to examine genetic variation and risk for common diseases.

The training data usually contains three types of information: the treatment given to the patient, the prognostic factors for the patient, and the outcome, some kind of measurement of the well-being of the patient. Dynamic treatment regimes are individually tailored treatments that are designed to provide treatment to individuals only when and if they need the treatment. Dynamic treatments explicitly incorporate the heterogeneity in need for treatment across individuals and the heterogeneity in need for treatment across time within an individual. Dynamic treatment regimes are attractive also because they only treat subjects who need them (see Murphy 2003). In treating cystic fibrosis, clinicians routinely update therapy according to the risk of toxicity and antibiotics resistance and hence adaptive treatment regimens work well here (Flume et al. 2007). This type of framework is also natural for cancer applications, where the initiation of the next line of therapy depends on the disease progression and thus the number of treatments is flexible. For example, in advanced nonsmall cell lung cancer (NSCLC), patients receive one to three treatment lines. The timing of the second and third lines of treatment is determined by the disease progression and by the ability of patients to tolerate therapy (see Stinchcombe and Socinski 2008, Krzakowski et al.

2010).

In a dynamic treatment regime, decision rules are specified before the beginning of treatment, and these rules are based on time-varying measurements of subject-specific needs. The set of decision rules comprises the treatment regime. A big challenge in identifying the optimal dynamic treatment regime (DTR) is that the optimal treatment sequence is unknown in the training data since the patients are given the treatments randomly. Incorporating patient information accrued over time into the decision rules is also challenging, and we also want to avoid treatments which may appear optimal in the short term, but may lead to poor final outcome in the long run. Censoring might be present as well due to loss to follow-up and hence the final outcome of those who reached the end of the study alive may be unknown. The number of decision points and the timing of these decision points can be different for different patients as well. All these challenges make estimating the effects of dynamic treatment regimes difficult. Nevertheless, it has been studied at length (see Robins 1993; 1997), and many approaches have been developed to optimally evaluate them since. One of the foremost methods to study the dynamic treatment regimes was formulated through the potential outcomes approach, that is, modeling the counter factual response observed by the patient if he/she had been assigned to a different treatment (see Rubin 1974, Robins 1986). The sequential multiple assignment randomized trial (SMART) designs (see for instance, Lavori and Dawson 2000; 2004, Murphy 2005a, Murphy et al. 2006, Moodie et al. 2007) was developed to relate the potential outcomes with the observed data. In this design, patients are randomized at every decision point, that is, the treatment assignments are independent of the future outcomes, conditional on the current history. Thus Murphy (2003)'s 'no unmeasured confounders' assumption, the essential condition guarantying the validity of the inferred optimal DTRs from the observed data, is satisfied.

A number of methods have been proposed to estimate the optimal DTRs. Lavori and Dawson (2000) proposed multiple imputation to estimate the potential outcomes and the best strategy is selected among all strategies by comparing their imputed outcomes. Murphy et al. (2001) proposed a structural mean response model to estimate the the unobserved latent responses for a particular DTR. Thall et al. (2000; 2002; 2007), proposed likelihood based methods incorporating both frequentist and Bayesian approaches to estimate parameters and thus the optimal regimes. Machine learning methods have been proposed as an alternative approach to estimating DTRs, and these methods have gained popularity due to their relatively model-free approach. The most important of them is Q-learning (Murphy et al. 2006, Zhao et al. 2009; 2011). The Q-learning algorithm, originally proposed by Watkins (see Watkins 1989, Watkins and Dayan 1992) in the computer science literature, has become a powerful tool to discover optimal DTRs in the clinical research arena. The Q learning algorithm can convert the multistage problem to a array of single stage problems, so that we can estimate the optimal rules sequentially in a pseudo-single stage setting. It finds the optimal decision rule at a given stage by first estimating the expectation of the sum of current and future rewards conditional on the current patient history assuming that the patient always receive optimal decisions at all future decision points, and then maximizing this empirical conditional expectation over the current set of decision rules. The estimated conditional expectations are known as Q-functions. Q learning is one of the most widely accepted methods to solve the reinforcement learning problem. Reinforcement learning is an area of machine learning in computer science, concerned with how an agent is supposed to take actions in an environment, so as to maximize some notion of cumulative reward. It is different from supervised learning which deals exclusively in learning from examples provided by a knowledgeable external supervisor. However, it alone is not adequate for learning from interaction. In interactive problems it is

often impractical to obtain examples of desired behavior that are both correct and representative of all the situations in which the agent has to act. In uncharted territory where one would expect learning to be most beneficial an agent must be able to learn from its own experience.

Finding therapies tailored to each individual in settings involving multiple decision times is a major challenge, and Q-learning can be used for maximizing the average survival time of patients as a function of prognostic factors, past treatment decisions, and optimal timing. Zhao et al. (2009) introduced the clinical reinforcement trial concept based on Q-learning for discovering effective therapeutic regimens in potentially irreversible diseases such as cancer. The concept is an extension and melding of dynamic treatment regimes in counter factual frameworks (Murphy 2003, Robins 2004) and sequential multiple assignment randomized trials (Murphy 2005a) to accommodate an irreversible disease state with a possible continuum of treatment options. This treatment approach falls under the general category of personalized medicine. The generic cancer application developed in Zhao et al. (2009) takes into account a drugs efficacy and toxicity simultaneously. The authors demonstrate that reinforcement learning methodology not only captures the optimal individualized therapies successfully, but is also able to improve longer-term outcomes by considering delayed effects of treatment.

In Q-learning, the optimal DTRs are estimated sequentially in a two-step procedure: the first step involves estimating the Q-functions at each stage using the prognosis history of the patient till that stage; and the next step involves maximizing these fitted functions over all the current decision rules to infer the optimal rule at that stage. One important problem that we typically face in this format is that the information about prognosis is sometimes very rich. And due to the Q-learning framework, this prognosis information (or history) grows with the number of stages in the trial. The effects of this redundant information in the history on which the Q functions are fitted are in fact

multifold. We can incur serious cost in collection and storage of this information but more importantly, this increases significantly the chances of overfitting. In presence of high-dimensional information, it is possible then that the Q-functions are poorly fitted or even grossly overfitted, and an overfitted model is not generalizable for predicting optimal treatments for future patients. In presence of noise or misspecification of the models for the Q functions, the fitted Q-functions may not necessarily result in maximal long-term clinical benefit. Like in any other learning problem, overfitting is an issue that needs to be addressed, and hence feature elimination is of significant importance in reinforcement learning frameworks and this is the primary focus for this part of the dissertation.

Feature selection hasn't been studied in great detail in the Q learning framework. The estimation phase in Q learning involves specifying a model for the Q functions and estimating them. The models for the Q functions can be specified parametrically, semiparametrically and even non-parametrically. Zhao et al. (2009) proposed two non parametric methods for the estimation phase of the Q learning algorithm, namely the support vector machines (regression) and extremely randomized trees. The advantage of using non parametric methods for estimation is that it lessens the scope for misspecification of the Q functions, in the presence of which it has been shown that the estimated DTRs may be suboptimal (see Murphy 2003). Here we have adopted the support vector machines framework with the Gaussian RBF kernel, which is sufficiently rich and produces an RKHS which is dense in the space of all continuous functions, and thus allow to capture any meaningful relationship the Q functions exhibit in the feature space satisfactorily. However as mentioned before, in presence of noise, there might still be significant amount of overfitting that may result in poor performance in prediction. Hence feature selection techniques can improve the performance of the Q learning algorithm sufficiently. The main contribution of this part of the dissertation is

to study three such methods for feature selection in Q learning. The idea stems from our earlier work (see Dasgupta et al. 2013), where we studied a backward feature selection method called the recursive feature elimination in support vector machines, and showed its generability in a variety of complex settings, when standard methods fail. We even showed that under certain regularity conditions, the method we proposed is consistent in finding the correct subset of features in these situations, thus establishing the usefulness of such a method in this regard.

The first method we study is the simple extension of our method in Q learning, using RFE at each estimation stage by finding a subset of features from the history variables at that stage. The second method introduced here uses a different version of the RFE algorithm that we propose here, called the RFE_test algorithm. It differs from the original RFE algorithm we proposed in (Dasgupta et al. 2013) in the criterion of deletion of the features. At each recursive step of the algorithm, the RFE algorithm calculates the change in the empirical regularized risk of the estimated SVM function after deletion of each of the features remaining in the model, and then it removes the feature that observes the lowest difference in the regularized empirical risk, thus performing an implicit ranking of features. In the original RFE algorithm, the risk estimates were obtained from the same data that was used for training the models. In this new proposed approach, we get our risk estimates from a separate data fold, that is, we employ a separate set of data for model building (training set) and then a separate set of data to evaluate the model performance (test set). The heuristics for the proposed modification comes from the observation that when the input dimension of the feature space is high compared to the number of signals in the model, it is likely that for the observed data, the model might overfit itself within the noisy dimensions satisfactorily, thus inflating the risk of elimination of the relatively weaker signals, while random variations in the data might be misclassified as important patterns.

The third method differ significantly from the first two methods, but in essence is a backward selection method as well. Two important differences include,

(1) Unlike the first two methods, this method works on the entire model building procedure, and not sequentially on each estimation phase.

(2) The model evaluation criterion is not regularized risk, but the optimal value function itself.

At each step of the feature selection procedure here, given the current size of the histories $(H_1, \ldots, H_T)$, we train the entire Q learning algorithm to obtain the empirical estimates of the stage 1 value function on submodels created sequentially by removing one feature at a time from the cumulative history, and then choosing the one that produces the highest estimate of the stage 1 value function for deletion. Heuristically this makes sense, as one of the main goals of the Q learning algorithm is to maximize the optimal stage 1 reward or value function.

1.4 Overview of the dissertation

In Chapter 2 we propose a statistically efficient way to handle medical adherence using temporal process regression. Chapter 3 is a detailed study of recursive feature elimination in support vector machines. In Chapter 4 we study different feature selection techniques in Q learning. And finally in Chapter 5, we summarize our findings in this dissertation and discuss future topics briefly.

CHAPTER 2: USING TEMPORAL REGRESSION TO STUDY ADHERENCE IN HEPATITIS C

2.1 Methods

In this section we review temporal process regression. We start off with a brief review of the model and assumptions, and explain how to produce pointwise confidence intervals and confidence bands for the processes. We also discuss how to use smoothing to getter better estimators of these processes and their confidence bands. Lastly we propose some hypothesis tests to test for the significance of the effects of these parametric processes in the given framework.

2.1.1 Model

Fine et al. (2004) proposed the following functional generalized linear model as an extension of standard linear models. The mean of the response $Y(t)$ at time t is specified conditionally on a $p \times 1$ vector of time-dependent covariates $X(t)$. That is,

$$\mu(t) = E\left(Y(t)|X(t)\right) = g^{-1}\{\beta(t)'X(t)\} \tag{2.1}$$

where the link function g is monotone, differentiable, and invertible, and $\beta(t) = \{\beta_1(t), \ldots, \beta_p(t)\}$ is a $p \times 1$ vector of time-dependent coefficients. The parameter $\beta(t)$ has a clear meaning in the model at time t and because the link is time-independent, $\beta(s)$ and $\beta(t)$ are comparable for $s \neq t$.

In our case, $Y_i(t)$ is constant for each patient i, and is the binary indicator for

whether that patient attained SVR at the end of the study. $X_i(t)$ is the covariate vector for patient i at time t (which includes adherence) and the link function is given as $g^{-1} = \exp/(1+\exp)$. Hence this actually gives us a time-indexed logistic model with $\beta(t)$ denoting the change in the log odds ratios for SVR per unit increase in the covariates at time t. Hence this can be interpreted as a cluster of generalized linear models, one for each time point t. We obtain estimates for the changes in the log odds ratios for SVR for different covariates for each time point t and these estimated effect sizes can be interpreted as processes varying over time. In practice, the data processes may be missing at some times. We only take into consideration those time points t where $\{Y(t), X(t)\}$ is fully observed at t.

Within a time interval $[l, u]$, we continuously observe n independent and identically distributed copies of $\{Y(t), X(t) : R(t) = 1\}$, where Y is the response, X is a $p \times 1$ covariate vector, and R is the data availability indicator, which permits both missing response and missing covariates. The estimator for $\beta(t)$ may be computed separately at each t. Denote $\hat{\beta}(t)$ as the root of $U\{\beta(t), t\} = \sum_{i=1}^{n} A_i(\beta(t), t)$, where

$$A_i\{\beta(t), t\} = R_i(t) D_i'\{\beta(t)\} V_i\{\beta(t), t\}[Y_i(t) - \mu_i(t)\}]$$

$D_i'\{\beta(t)\} = \partial[g^{-1}\{\beta(t)' X_i(t)\}]/\partial\{\beta(t)\}$ and $V_i\{\beta(t), t\}$ is a weight matrix possibly random.

The estimator potentially jumps at those M times where $\{Y_i(t), X_i(t) : R_i(t) = 1\}$ and $R_i(t)$ jumps. Let $j_1 < \cdots < j_M$ be the jump points. Finding $\hat{\beta}(t)$ involves solving $U\{\beta(t), t\}$ at the M points. According to Fine et al. (2004), in theory, when the processes vary between j_i, smoothing is not required. But in practice, the equations are solved on a grid and the estimators are interpolated via smoothing. If $Y_i(t)$ and $X_i(t)$ are piecewise constant, then so is the estimator.

2.1.2 Pointwise Confidence Intervals

Under appropriate conditions (See Liang and Zeger (1986)), for each t, $\hat{\beta}(t)$ is consistent for $\beta_0(t)$, the true value of $\beta(t)$ and for any $K < \infty$ points with $l < t_1 < \cdots < t_K < u$, $n^{1/2}[\{\hat{\beta}(t_1)', \ldots, \hat{\beta}'(t_K)\}' - \{\beta_0(t_1)', \ldots, \beta_0(t_K)'\}]$ is asymptotically normal with covariance consistently estimated by the sandwich estimator.

Hence pointwise confidence intervals for $\beta_0(t)$ may be constructed using the normal approximation and the sandwich variance estimate

$$\hat{\Sigma}(t,t) = \{\hat{H}(t)^{-1}\}\hat{G}(t,t)\{\hat{H}(t)^{-1}\}',$$

where $\hat{H}(t)$ and $\hat{G}(t,s)$ are given by

$$\hat{H}(t) = n^{-1}\sum_{i=1}^{n} R_i(t)D_i'\{\hat{\beta}(t)\}V_i\{\hat{\beta}(t),t\}D_i\{\hat{\beta}(t)\},$$

$$\hat{G}(s,t) = n^{-1}\sum_{i=1}^{n} A_i\{\hat{\beta}(s),s\}A_i\{\hat{\beta}(t),t\}'.$$

A $100(1-\alpha)\%$ confidence interval at time t for $\beta_{k0}(t)$ is $\hat{\beta}_k(t) \pm n^{-1/2}z_{\alpha/2}\hat{\Sigma}_k(t,t)^{1/2}$ where $z_{\alpha/2}$ is the $(1-\alpha/2)$ percentile of the standard normal distribution and $\hat{\Sigma}_k(t,t)$ is the k^{th} diagonal element of $\hat{\Sigma}(t,t)$.

2.1.3 Confidence Bands

In Fine et al. (2004), the authors show that the consistency and weak convergence results hold uniformly in t, that is, $\hat{\beta}(t)$ converges uniformly to $\beta_0(t)$ for $t \in [l,u]$ and $n^{1/2}\{\hat{\beta}(t) - \beta_0(t)\}$ converges weakly to a tight zero mean Gaussian process $\mathcal{G}(\cdot)$ with continuous sample paths at continuity points of $\beta_0(t)$ with the covariance function $\Sigma(s,t) = \left[n^{1/2}\{\hat{\beta}(t) - \beta_0(t)\}, n^{1/2}\{\hat{\beta}(t) - \beta_0(t)\}\right] = \{H(s)^{-1}\}G(s,t)\{H(t)^{-1}\}'$, where $G(s,t)$ and $H(t)$ are asymptotic limits of $\hat{G}(s,t)$ and $\hat{H}(t)$ respectively.

However, constructing confidence bands for $\beta(t)$ for $t \in [l, u]$ is analytically difficult since the Gaussian process $\mathcal{G}(\cdot)$ does not have a canonical representation. Instead, we can employ resampling, either by bootstrapping the empirical data distribution and solving $U\{\beta(t), t\}$ repeatedly, or by simulating directly from the process, as in Lin, Fleming, and Wei (1994). For a better understanding of this, one may refer to Chapter 22, Section 3 of Kosorok (2008) which studies this particular case in detail. In this analysis we however use a conservative approach. We sample from the empirical distribution of a standardized version of $\sup_t |\hat{\beta}(t) - \beta_0(t)|$. This results in wider confidence bands. The importance of this approach lies in the fact that we can obtain a direct correspondence between these confidence bands and hypotheses tests which will be explained later. The way we generate these confidence bands is as follows:

We use the fact that $n^{1/2}\{\hat{\beta}(t) - \beta_0(t)\} = n^{-1/2} \sum_{i=1}^{n} \psi_i(t) + o_p^t(1)$ where

$$\psi_i(t) = \{H(t)\}^{-1} A_i\{\beta_0(t), t\}$$

is the influence function for the process $\hat{\beta}(t)$. Note that the sandwich variance estimator is given by $\hat{\Sigma}(s, t)$ as before. Now we can define $\hat{\overline{\psi}}_i(t)$ as

$$\hat{\overline{\psi}}_i(t) = [\text{diag}(\hat{\Sigma}(t, t))]^{-1/2}\{\hat{H}(t)\}^{-1} A_i\{\hat{\beta}(t), t\}.$$

Hence we can create $100(1 - \alpha)\%$ simultaneous confidence bands of the form

$$\hat{\beta}_k(t) \pm n^{-1/2} b_{k,\alpha/2} \hat{\Sigma}_k(t, t)^{1/2}, \tag{2.2}$$

where $b_{k,\alpha/2}$ is the $(1 - \alpha/2)$-th quantile of the empirical distribution of the k^{th} component of B_n where,

$$B_n = \sup_t \left\{ n^{-1/2} \left| \sum_{i=1}^{n} z_i \hat{\overline{\psi}}_i(t) \right| \right\}$$

from repeatedly sampling $z_1, \ldots, z_n \sim$ i.i.d. $N(0, 1)$.

2.1.4 Smoothing the estimated parametric function

Smoothing is a class of regression techniques to estimate a real valued function $f(X)$ over the domain $\mathbb{R}$ by using its noisy observations, and fitting a different but simple model separately at each query point x_0. This is done by using observations close to the target point x_0 to fit the simple model, in such a way that the resulting estimated function is smooth in $\mathbb{R}$.

Here we assign weights that die off smoothly with distance from the target point. For each t_0, the Nadaraya-Watson kernel-weighted average is defined as

$$\tilde{\hat{\beta}}(t_0) = \frac{\sum_{i=1}^{n} K_\lambda(t_0, t_i)\hat{\beta}(t_i)}{\sum_{i=1}^{n} K_\lambda(t_0, t_i)}.$$

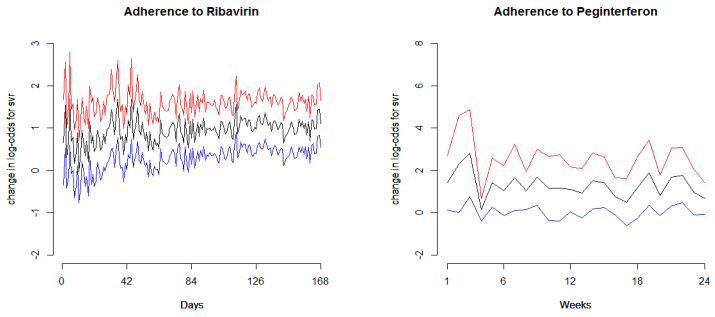
We use the triangular kernel for smoothing, defined as

$$K_\lambda(t_0, t) = D\left(\frac{|t_0 - t|}{\lambda}\right),$$

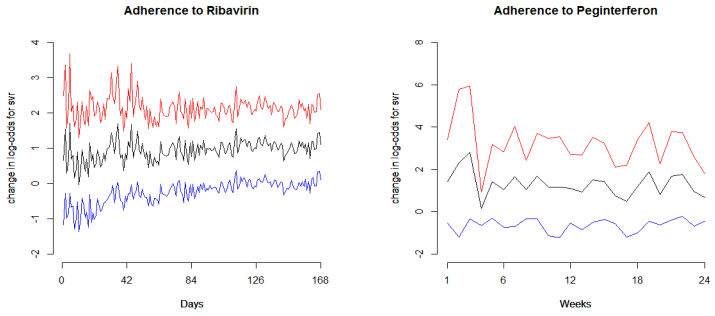
where

$$\begin{aligned} D(t) &= (1 - t) \quad \text{if } |t| < 1 \\ &= 0 \quad \text{otherwise} \end{aligned}$$

The amount of smoothing that we want can be controlled by the kernel width λ, where λ is typically chosen using cross validation.



(A)



(B)

Figure 2.1: (A) 95% Pointwise Confidence Intervals for effects of adherence (compliance) on the log odds for SVR for the first 24 weeks in separate analyses of the two drugs. (B) 95% Confidence Band for effects of adherence (compliance) on the log odds for SVR for the first 24 weeks in separate analyses of the two drugs.

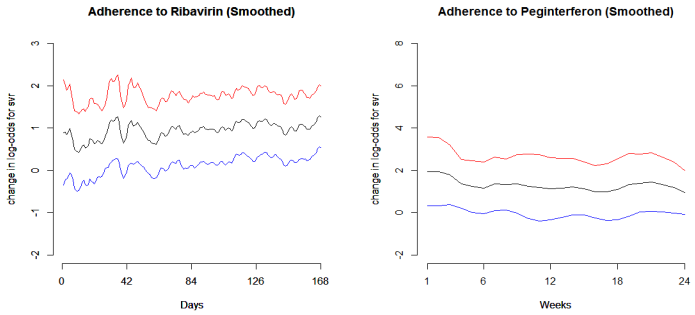


Figure 2.2: 95% Confidence Band for effects of adherence (compliance) on the log odds for SVR for the first 24 weeks in separate analyses of the two drugs smoothed over time.

2.1.5 Confidence bands using the smoothed estimators

As before, we can produce confidence bands for the parametric processes using the smoothed versions of the estimated parametric function.

We have

$$n^{1/2}\{\hat{\beta}(t) - \beta_0(t)\} = n^{-1/2}\sum_{i=1}^{n}\psi_i(t) + o_p^t(1),$$

where $\psi_i(t)$ is defined as before. Now if $\hat{\psi}_i(t)$ is the estimated influence function, then define $\tilde{\hat{\psi}}_i(t)$ to be the smoothed version of $\hat{\psi}_i(t)$. Hence we can produce smoothed version of the Sandwich variance estimator as $\tilde{\hat{\Sigma}}(t,t) = n^{-1}\sum_{i=1}^{n}\left[\tilde{\hat{\psi}}_i(t)\right]\left[\tilde{\hat{\psi}}_i(t)\right]'$. Now define $\tilde{\bar{\psi}}_i(t)$ as $\tilde{\bar{\psi}}_i(t) = [\text{diag}\tilde{\hat{\Sigma}}(t,t)]^{-1/2}\tilde{\hat{\psi}}_i(t)$, and hence we can create $100(1-\alpha)\%$ smoothed simultaneous confidence bands of the form

$$\tilde{\hat{\beta}}_k(t) \pm n^{-1/2}\tilde{b}_{k,\alpha/2}\tilde{\hat{\Sigma}}_k(t,t)^{1/2},$$

where $\tilde{b}_{k,\alpha/2}$ is the $(1-\alpha/2)$-th quantile of the empirical distribution of the k^{th} component of $\tilde{B}_n$ where,

$$\tilde{B}_n = \sup_t\left\{n^{-1/2}\left|\sum_{i=1}^{n} z_i\tilde{\bar{\psi}}_i(t)\right|\right\}$$

from repeatedly sampling $z_1,\ldots,z_n \sim$ i.i.d. $N(0,1)$. This follows from the continuous mapping theorem.

2.1.6 Non-parametric hypothesis tests

Fine et al. (2004) proposed three different non parametric tests for testing the null hypothesis $H_0 : C(t)\beta(t) = c(t)$, where at each t, $C(t)$ is an $r \times p$ contrast matrix and $c(t)$ is an $r \times 1$ vector of constants. This general framework allows global tests for multiple hypotheses. In this analysis, we consider only two of them. The first statistic is an integrated difference statistic (IDS). Defining $M(t) := C(t)\hat{\beta}(t) - c(t)$ we have,

$$T_1 = \int_l^u M(t)W(t)dt,$$

where W is a non-negative weight function, possibly random. Under mild conditions $T_1'\hat{\Sigma}_1^{-1}T_1$ is asymptotically χ_r^2 under H_0, where

$$\hat{\Sigma}_1 = n^{-2}\sum_{i=1}^{n}\left(\int_l^u C(s)\hat{H}(s)^{-1}A_i\{\hat{\beta}(s),s\}W(s)ds\right)^{\otimes 2},$$

and for a vector v, $v^{\otimes 2} = vv'$. The second statistic is the supremum difference statistic (SDS), based on the sup-norm distance,

$$T_2 = \sup_{t\in[l,u]}\left|M(t)'\Big\{C(t)\hat{\Sigma}(t,t)C(t)\Big\}^{-1}M(t)\right|.$$

Similarly to most Kolmogorov-Smirnov type statistics, the distribution of T_2 is rather complex and is typically approximated by resampling.

A simple test of $\beta_j = 0$ can be visually determined by looking at the confidence band of β_j and determining whether at any time point the whole portion of the band lies above or below 0.

2.2 Results

Our primary focus is the first 24 weeks of treatment. And our aim is to analyze the effect of adherence to Ribavirin and Peginterferon on the outcome sustained virologic response (SVR). For simplicity, we model Ribavirin as a binary predictor, and hence create a pseudo score, where no or partial adherence to the drug is given the score 0 while full adherence is given the score 1. We start off with an initial analysis where we model adherence to the two drugs separately. We give plots and use the proposed tests to test for their significance. We also look for other covariates that affect SVR significantly in this set up. We then conduct a combined analysis of adherence to the drugs in a single framework and look for substantial effects of interaction between them. Based on our results, we conduct a few additional analyses to look for meaningful

conclusions.

2.2.1 Initial Plots

We begin by using temporal process regression to model Ribavirin and Peginterferon separately. Figure 2.1(A) shows the estimated effect sizes for adherence on SVR in the two analyses. Hence for the Ribavirin analysis, the estimated effect size $\beta(t)$ for the t^{th} day (as seen in the plot) is the log odds for an individual under complete compliance with Ribavirin on the t^{th} day, to attain SVR over an individual under partial or no compliance with the drug on that day. Similarly for the Peginterferon analysis, the estimated effect size for the i^{th} week is the log odds for an individual under compliance with Peginterferon on the i^{th} week, to attain SVR over an individual under no compliance with the drug during that week. In Figure 2.1(A), we also plot the 95% pointwise confidence intervals for these processes. Figure 2.1(B) is a plot of the 95% confidence bands for the change in log odds for SVR under complete compliance for the two drugs. As expected, the confidence bands are wider than pointwise confidence intervals.

As is evident from Figures 2.1(A) and 2.1(B), the estimated processes are quite noisy, since we estimate the effects on a daily or a weekly grid (depending on the drug we are analyzing) and interpolate over rest of the interval. Hence to obtain a better estimate of the processes, we employ kernel smoothing (refer to Section 2.1.4). The results are presented in Figure 2.2. We provide the estimated processes smoothed over time along with their 95% confidence bands.

2.2.2 Results of Non Parametric Hypothesis Tests

Results for T_1 (IDS): Separately, adherence to Ribavirin ($p = 8.413 \times 10^{-7}$) and adherence to Peginterferon ($p = 0.000473$) are found to be highly associated with SVR

by the integrated difference statistic.

Results for T_2 (SDS): As we see in Figure 2.2(D), the lower confidence bands cross 0 indicating that the effects are found to be positively significant for both Ribavirin and Peginterferon by the supremum difference statistic.

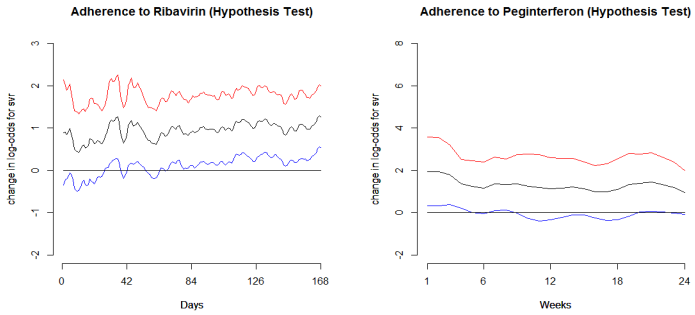


Figure 2.3: Plots for Hypothesis test T_2.

2.2.3 Backward Selection of Covariates

We fit a full model considering other predictors, and follow a gradual step down procedure to remove the ones which weren't found to be significantly associated with SVR, after controlling for the other predictors. The covariates considered for the full model are listed below:

- SEX : Gender
- RACEW: Whether caucasian/african-american
- MXAD: History of anti-depressant use
- Age: Age of the subject
- ISHAK: Indicator of severity of disease (fibrosis score)
- Infect: Source of infection

- Education: Education level of the subject
- Insurance: Insurance provider for the subject
- Employ: Employment Status
- Marital: Marital Staus
- Alcohol: Alcoholic Status
- Vload: Baseline Viral Load Score

A flow chart of the Backward Selection procedure is given in Figure 2.4. The steps are performed for analyses of both drugs. The final models after the step down process

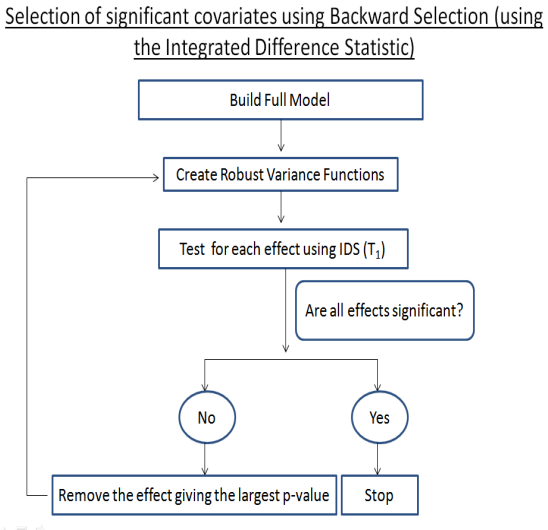


Figure 2.4: Backward Selection Procedure for choosing the significant covariates in the model.

are found to consist of the same covariates and are shown in Table 2.1.

2.2.4 Plots for other significant covariates

It is clear that adherence (or compliance) to the drug regimes is extremely important and is positively associated with log odds for SVR. Table 2.1 shows that apart from

Covariates	Ribavirin Analysis	Peginterferon Analysis
SEX	0.029	0.016
RACE	4.71e-05	2.81e-05
ISHAK	0.009	0.009

Table 2.1: Final Model P-Values

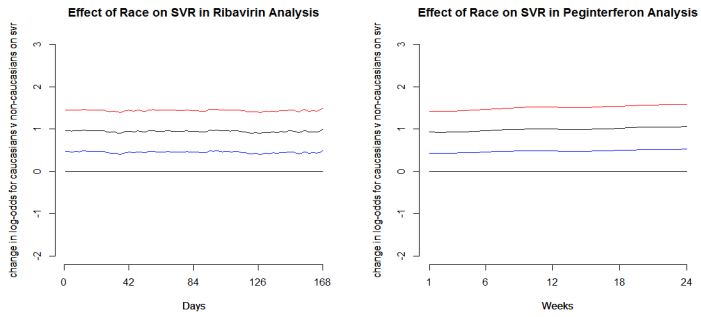


Figure 2.5: Effect of Race = Caucasian on the log odds for SVR for the first 24 weeks in separate analyses of the two drugs.

adherence, SEX, RACEW and ISHAK are significantly associated with SVR too. We plot the estimated effects in Figures 2.5–2.7.

Note that the plots are almost straight lines which is expected since these covariates are constant over time. Figure 2.5 shows that race (= Caucasian) is positively associated with SVR which means that Caucasian patients are significantly more likely to attain SVR than non-Caucasians. Figure 2.6 shows that gender (= Male) is negatively associated with SVR which means that Female patients are significantly more likely to attain SVR than Males. And lastly Figure 2.7 shows that fibrosis score is negatively associated with SVR. The fibrosis score denotes the severity of the disease so it makes sense that more severe cases of Hepatitis C have a significantly lower probability of attaining SVR.

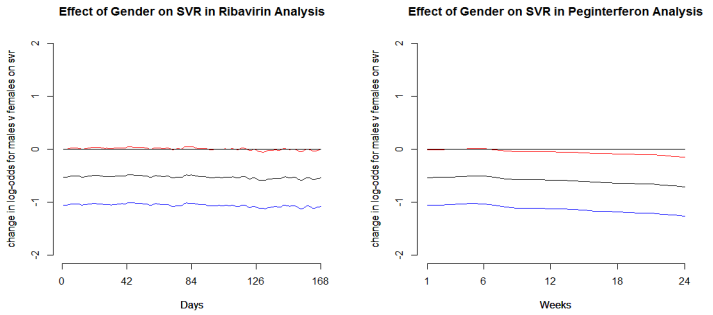


Figure 2.6: Effect of Gender = Male on the log odds for SVR for the first 24 weeks in separate analyses of the two drugs.

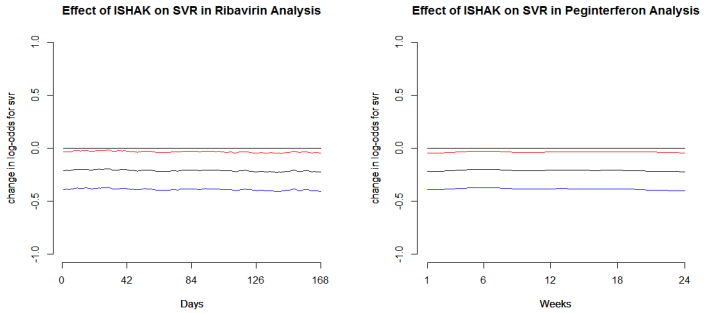


Figure 2.7: Effect of Fibrosis Score on the log odds for SVR for the first 24 weeks in separate analyses of the two drugs.

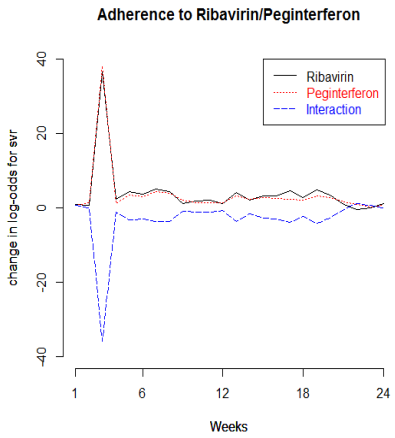


Figure 2.8: Plot for the main effects and interaction of adherence (compliance) to the drugs on SVR

2.2.5 Combined Analysis

Since the prescribed regimen is really a combination of the two drugs, we now conduct an analysis on the combined Ribavirin-Peginterferon data. We first incorporate only the fixed effects for adherence to the drugs (Peginterferon and Ribavirin) and the covariates (SEX, RACEW and ISHAK) found significant in the separate analyses. Since Peginterferon is taken once every week, the analysis is done across weeks. The daily information on the Ribavirin drug is introduced as a score vector of length 7 for each week, with the i^{th} element recording the score for i^{th} day of the week ($i = 1, \ldots, 7$). The first hypothesis that we test is $H_0 : \beta_1 = \cdots = \beta_7$, where the parameter β_i represents the effect of adherence to Ribavirin on SVR for the i^{th} day of the week. Hence we test whether the effect of adherence to drug Ribavirin on SVR is the same across different days of a week. Both the Integrated Effect Test T_1 (p = 0.863) and the Supremum Effect Test T_2 (p = 0.561) showed lack of sufficient evidence against the null hypothesis, meaning that the Ribavirin adherence can be adequately summarized by the weekly average.

Accordingly, we now create a single covariate for adherence to Ribavirin for each week by taking the average of the daily scores for each week. We then test for the significance of adherence to the drugs in the same model. The integrated difference statistic shows that the individual effects of the drugs Ribavirin (p = 0.0202) and Peginterferon (p = 0.009) are still both significant, though the more conservative supremum difference statistic did not find sufficient evidence at 5% level of cut-off to support that (p 0.258 and 0.081 for Ribavirin and Peginterferon respectively). However both tests, IDS and SDS found the joint effect of the drugs to be highly significant (p = 0.00026 and 0.007 respectively).

As the next logical step, we introduce an interaction term in this combined analysis (the effect of interaction between the two drugs Ribavirin and Peginterferon on the

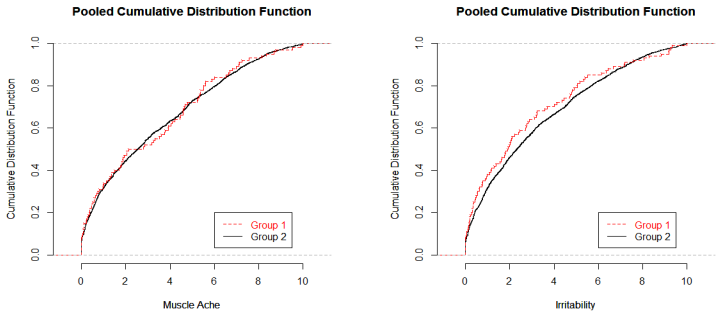
change of log odds for SVR). We make a very interesting observation from the analysis as seen in Figure 2.8, where we see huge peaks in the estimated main effects of adherence for the two drugs, and a huge dip in the estimated interaction effect on week 3. On further investigation of SVR on week 3, we realize that there is a perfect separation of the data based on the interaction of the two drugs Ribavirin and Peginterferon. All of the subjects who were non-adherent to Peginterferon for that week and were at best partially adherent to Ribavirin for the whole week, didn't show Sustained Virologic Response at the end of the study. Also interestingly, only 1 among 22 patients (4.55%) who were non-adherent to Peginterferon for that week showed Sustained Virologic Response at the end of the study, while the percentage of SVR among those who did adhere to Peginterferon on week 3 was 41.11%. This calls for further analysis on these 22 patients who failed to adhere to the drug Peginterferon on week 3.

2.2.6 Diagnostic analysis

On week 3, 22 patients did not adhere to Peginterferon (group 1) while the remaining individuals did (group 2) adhere to Peginterferon. We want to compare these two groups with respect to several criterion scores, both physical and physiological. The physical scores include various symptom scores, (i) Muscle Ache, (ii) Irritability, (iii) Headache, (iv) Fatigue, (v) Depression, and (vi) Overall. And the physiological scores include various internal measurements such as, (i) WBC count, (ii) NPC count, (iii) Platelet count, and (iv) Viral Load scores.

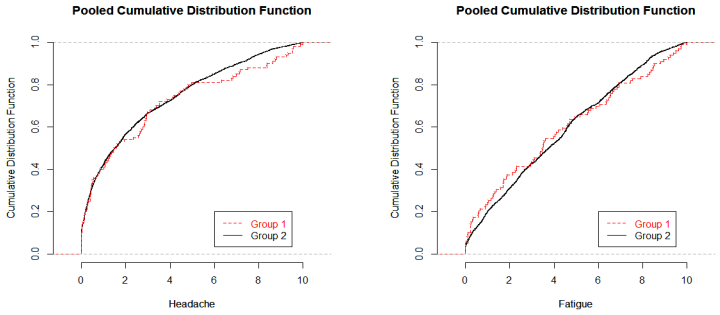
In Figures 2.9 – 2.13, we look at the cumulative distribution plots of these scores, pooled across the entire length of study.

The only interesting plot is for the viral load scores in Figure 2.13(B). The lower curve representing group 1 shows that the viral load scores of group 1 tend to have higher values than that of group 2 as we would expect. We use the Cramer Von-Mises



(A) (B)

Figure 2.9: Plots for the pooled (across weeks) cdf for the two groups (group 1 consist of patients not adhering to Peginterferon on week 3, while group 2 consist of the remaining patients who did adhere to the drug during that week) for the physical scores: (A) muscle ache, and (B) irritability.

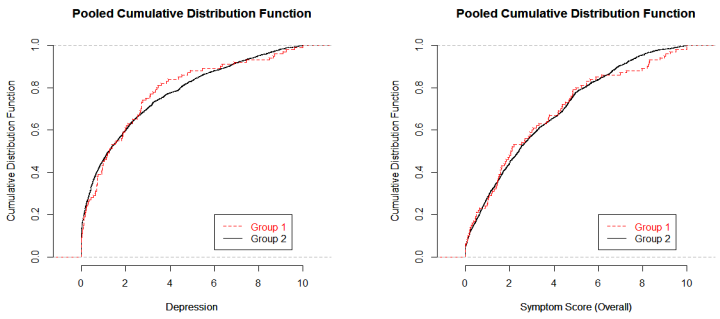


(A) (B)

Figure 2.10: Plots for the pooled (across weeks) cdf for the two groups (group 1 consist of patients not adhering to Peginterferon on week 3, while group 2 consist of the remaining patients who did adhere to the drug during that week) for the physical scores: (A) headache, and (B) fatigue.

criterion to test whether these differences are significant, where the null distribution is simulated using bootstrap samples from the data itself to adjust for repeated measures.

As expected from the plots, the Cramer Von-Mises criterion did not bear any evidence to reject the null hypothesis of no difference in distribution, for all the attributes



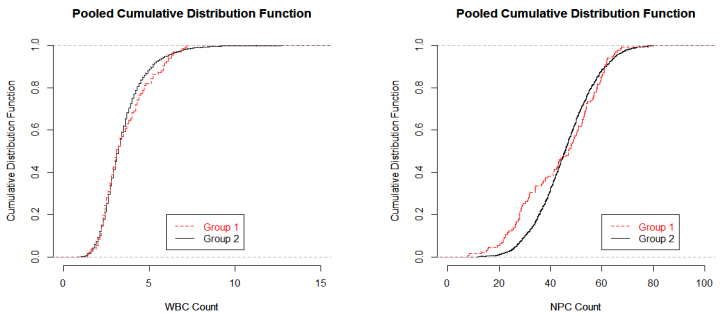
(A) (B)

Figure 2.11: Plots for the pooled (across weeks) cdf for the two groups (group 1 consist of patients not adhering to Peginterferon on week 3, while group 2 consist of the remaining patients who did adhere to the drug during that week) for the physical scores: (A) depression, and (B) overall symptom scores.

except the viral load scores. A 5000 simulation run gave a p. value of 0.0094, demonstrating that the distribution of the pooled viral load scores for the two groups are significantly different. Viral Load scores being a response criterion for our study indicates that early adherence to Peginterferon is extremely important. In Table 2.2, we give results from the Cramer von Mises test on the difference in viral load scores between the two groups on individual readings. The Cramer von Mises criterion showed non-significant results for the initial two time points. At the 3^{rd} time point (week 2) it approaches statistical significance and is significant by week 4.

2.3 Summary of Chapter 2

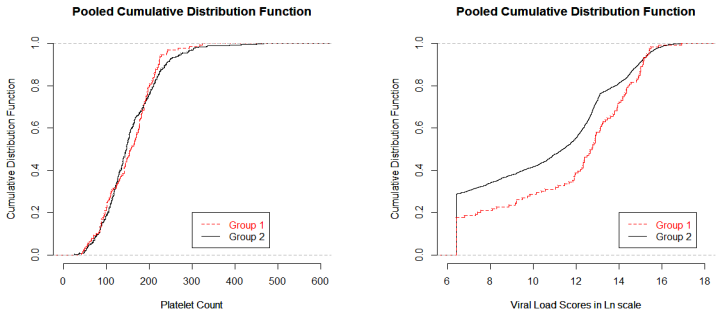
The initial analyses showed that adherence to both drugs has a significant effect on the treatment end-point (SVR), with higher adherence significantly increasing the chance of achieving SVR. This confirms the fact that adherence is crucial for effectiveness of the medication regimen for treating chronic hepatitis C. We also found other significant factors that affect SVR. It was seen that women have higher probability of



(A) (B)

Figure 2.12: Plots for the pooled (across weeks) cdf for the two groups (group 1 consist of patients not adhering to Peginterferon on week 3, while group 2 consist of the remaining patients who did adhere to the drug during that week) for the physiological scores: (A) WBC counts, and (B) NPC counts.

attaining SVR than men. We also saw that race plays an important role in determining chances for a positive drug response and that Caucasians have significantly higher chances of attaining SVR than others. We further saw that the severity of infections (fibrosis score) does affect SVR and patients with higher baseline infection scores have less chances of a full recovery (this reaffirms results found in Conjeevaram et al. (2006)). The combined analysis showed some interesting results as well. The individual effects of the drugs were found significant by the IDS test while the joint effect was found significant by both the IDS and SDS tests. This shows that adherence to the combined regimen is important to improve chances of achieving SVR, confirming results obtained from the Phase-II drug trials. Figure 2.8 showed that the effect of interaction between adherence to the drugs can also have a serious impact on SVR. Our results showed that adherence on week 3 has tremendous bearing on the final outcome, which supports the conclusion that adherence in the first few weeks of the regimen is extremely important.



(A) (B)

Figure 2.13: Plots for the pooled (across weeks) cdf for the two groups (group 1 consist of patients not adhering to Peginterferon on week 3, while group 2 consist of the remaining patients who did adhere to the drug during that week) for the physiological scores: (A) platelet counts, and (B) viral load scores.

Vload Score Reading	Cramer V Mises P-value
Day 1	0.2765
Week 1	0.2208
Week 2	0.0769
Week 4	0.0030
Week 8	0.0833
Week 12	0.1932

Table 2.2: Viral Load score difference P-Values across Weeks

CHAPTER 3: CONSISTENCY RESULTS FOR RECURSIVE FEATURE ELIMINATION IN SVM

3.1 Preliminaries

We begin by introducing some notations and discussing support vector machines (and empirical risk minimization as a related concept).

Let the input space $(\mathcal{X}, \mathcal{A})$ be measurable, such that $\mathcal{X} \subseteq B \subset \mathbb{R}^d$, where B is an open Euclidean ball centered at 0. Let $\mathcal{Y}$ be a closed subset of $\mathbb{R}$ and P be a measure on $\mathcal{X} \times \mathcal{Y}$. A function $L : \mathcal{X} \times \mathcal{Y} \times \mathbb{R} \mapsto [0, \infty]$ is called a loss function if it is measurable. We say that a loss function is convex if $L(x, y, \cdot)$ is convex for every $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. A loss function is called locally Lipschitz continuous with Lipschitz local constant $c_L(\cdot)$ if for every $a > 0$,

$$\sup_{x \in \mathcal{X}, y \in \mathcal{Y}} |L(x, y, s) - L(x, y, \acute{s})| < c_L(a) \left| s - \acute{s} \right|, \quad s, \acute{s} \in [-a, a].$$

L is said to be Lipschitz continuous if there is a constant c_L such that $c_L(a) \leq c_L$ $\forall a \in \mathbb{R}$.

For any measurable function $f : \mathcal{X} \mapsto \mathbb{R}$, we define the L-risk of f with respect to the measure P as $\mathcal{R}_{L,P}(f) = E_P[L(X, Y, f(X)]$. The Bayes Risk $\mathcal{R}^*_{L,P}$ is defined as $\inf_f \mathcal{R}_{L,P}(f)$, where the infimum is taken over the set of all measurable functions, $\mathcal{L}_0(\mathcal{X}) = \{f : \mathcal{X} \mapsto \mathbb{R}, \ \ f \text{ is measurable}\}$. A function f^*_P that achieves this infimum is called a Bayes decision function.

Let $\mathcal{F} \subseteq \mathcal{L}_0(\mathcal{X})$ be a non-empty functional space, and L be any loss function. Let

$$f_{P,\mathcal{F}} = \arg\min_{f \in \mathcal{F}} E_P[L(X, Y, f(X)] = \arg\min_{f \in \mathcal{F}} \mathcal{R}_{L,P}(f) \tag{3.1}$$

be the minimizer of infinite-sample risk within the space $\mathcal{F}$. We define the minimal risk within the space $\mathcal{F}$ as $\mathcal{R}^*_{L,P,\mathcal{F}} = \mathcal{R}_{L,P}(f_{P,\mathcal{F}})$. The empirical risk is denoted by $\mathcal{R}_{L,D}$ (where the subscript D denotes the empirical measure invoked by the data $D = \{(X_1, Y_1), \ldots, (X_n, Y_n)\} \in (\mathcal{X} \times \mathcal{Y})^n$), and is given as, $\mathcal{R}_{L,D}(f) \equiv \mathbb{P}_n(L(X, Y, f(X)) = \frac{1}{n}\sum_{i=1}^{n} L(X_i, Y_i, f(X_i))$.

Empirical Risk Minimization: A learning method whose decision function $f_{D,\mathcal{F}}$ minimizes empirical risk $\mathcal{R}_{L,D}(f)$ among the class of functions $\{f \ : f \in \mathcal{F}\}$, for all $n \geq 1$ and data D is called *empirical risk minimization (ERM)* with respect to L and $\mathcal{F}$.

Now let H be an $\mathbb{R}$-Hilbert space over $\mathcal{X}$. A function $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is called a *reproducing kernel* of H if $k(\cdot, x) \in H$ for all $x \in \mathcal{X}$, and has the reproducing property $f(x) = \langle f, k(\cdot, x)\rangle$ for all $f \in H$, and all $c \in \mathcal{X}$. The space is called a real-valued *Reproducing Kernel Hilbert Space (RKHS)* over $\mathcal{X}$ (For a better understanding of RKHSs, we refer our readers to SC08).

Support Vector Machines: Let H be a separable RKHS of a measurable kernel k on $\mathcal{X}$, and fix a $\lambda > 0$. Let L be convex and locally Lipschitz continuous. Then the *empirical SVM decision function* can be defined as,

$$f_{D,\lambda,H} = \arg\min_{f \in H} \lambda \|f\|^2_H + \mathcal{R}_{L,D}(f). \tag{3.2}$$

For a given λ, the SVM learning method $\mathfrak{L}$ is the map $(\mathcal{X} \times \mathcal{Y})^n \times \mathcal{X} \mapsto \mathbb{R}$ defined by $(D, x) \mapsto f_{D,\lambda,H}(x)$ for all $n \geq 1$. Like before, we can define the infinite sampled version of the regularized minimizer as $f_{P,\lambda,H} = \arg\min_{f \in H} \lambda \|f\|^2_H + \mathcal{R}_{L,P}(f)$. Then the

approximation error is given by,

$$A_2^H(\lambda) = \lambda \left\| f_{P,\lambda,H} \right\|_H^2 + \mathcal{R}_{L,P}\left(f_{P,\lambda,H}\right) - \inf_{f \in H} \mathcal{R}_{L,P}(f). \tag{3.3}$$

NOTE: The results developed in this paper are valid not only for classification, but also for regression under certain general assumptions on the output space $\mathcal{Y}$. For simplicity however, we will refer to both these variants in this paper as SVM, unless otherwise mentioned.

3.2 Feature Elimination Algorithm

The original recursive feature elimination (RFE) algorithm was proposed for SVMs by Guyon et al. (2002), the performance of which was evaluated under experimental settings. Limitations of this method as a margin-maximizing feature elimination was studied explicitly in Aksu et al. (2010). The version proposed here is similar in structure to Guyon et al., but differ in the elimination criterion. While Guyon et al. used the Hilbert space norm $\lambda \| f \|_H^2$ to eliminate features recursively, we use the entire objective function (the regularized empirical risk) for deletion.

3.2.1 The Algorithm

We begin by proposing a way such that starting off with a space $\mathcal{F}$, we are able to create lower dimensional versions of it. As mentioned before, this is indeed necessary, since at each stage of the feature elimination process, we move down to a 'lower dimensional' feature space and the functional spaces need to be adjusted to cater to the appropriate version of the problem in these subspaces. A detailed discussion on these will be given in Section 3.3.

Definition 1. *For any set of indices $J \subseteq \{1, 2, .., d\}$ and a given functional space $\mathcal{F}$,*

define $\mathcal{F}^J = \{g : g = f \circ \pi^{J^c} \quad , \forall f \in \mathcal{F}\}$, *where* π^{J^c} *is the projection map from* $x \mapsto x^J$ $(x, x^J \in \mathbb{R}^d)$, *such that* x^J *is produced from* x *by replacing those elements in* x *which are indexed in the set* J, *by zero.*

We can hence define the space $\mathcal{X}^J = \{\pi^{J^c}(x) \, : \, x \in \mathcal{X}\}$, such that $\pi^{J^c} : \mathcal{X} \mapsto \mathcal{X}^J$ is a surjection. Now we are ready to provide the algorithm. Assume the support vector machine framework, where we are given an RKHS H indexed by a kernel k.

Algorithm 2. *Start off with* $J \equiv [\cdot]$ *empty and let* $Z \equiv [1, 2, ..., d]$.

STEP 1: In the k^{th} *cycle of the algorithm choose dimension* i_k *for which*

$$\begin{aligned} i_k = & \underset{i \in Z \setminus J}{\arg\min} \, \lambda \left\| f_{D,\lambda,H^{J\cup\{i\}}} \right\|^2_{H^{J\cup\{i\}}} + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J\cup\{i\}}}\right) \\ & - \lambda \left\| f_{D,\lambda,H^J} \right\|^2_{H^J} - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right). \end{aligned} \tag{3.4}$$

STEP 2: Update $J = J \cup \{i_k\}$. *Go to STEP 1.*

Continue this until the difference $\min_{i \in Z \setminus J} \lambda \left\| f_{D,\lambda,H^{J\cup\{i\}}} \right\|^2_{H^{J\cup\{i\}}} + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J\cup\{i\}}}\right) - \lambda \left\| f_{D,\lambda,H^J} \right\|^2_{H^J} - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right)$ *becomes larger than a pre-determined quantity* δ_n, *and output* J *as the set of indices for the features to be removed from the model.*

See Appendix B.1.1 for a version of the algorithm for empirical risk minimization problems.

3.2.2 Cycle of RFE

We define 'cycle' of the RFE algorithm as the number of 'features' deleted in one step of the algorithm. The algorithms in 3.2.1 has cycle = 1. But one can define it for cycles of value greater than 1 in which case one deletes chunks of features at a time, equal to the size of the cycle. It can also be defined adaptively such that in different

runs of the algorithm the cycle sizes are different. The theoretical results derived in this paper will hold for cycles of any size. Here for the sake of simplicity, we set the cycle size to 1.

3.3 Functional Spaces on Lower Dimensional Domains

The aim of this section is to provide a detailed reasoning behind Definition 1 in Section 3.2.1.

3.3.1 Feature Elimination in SVM

In empirical risk minimization problems our primary focus is empirical risk $\mathcal{R}_{L,D}(f)$, while in support vector machines our main concern is the regularized version of this risk, $\lambda\|f\|_{\mathcal{F}}^2 + \mathcal{R}_{L,D}(f)$. The minimization in case of SVMs is typically computed over special functional classes called RKHSs (denoted by H here). Our objective is then to find $f_{D,\lambda,H} \equiv \arg\min_{f\in H} \lambda\|f\|_H^2 + \mathcal{R}_{L,D}(f)$. The regularization term $\lambda\|f\|_H^2$ is used to penalize functions f with a large RKHS norm. Complex functions $f \in H$ which model the output values in the training data set D too closely, tend to have larger H-norms (Refer to Exercise 6.7 in SC08 for a clear motivation).

Now consider the setting of empirical risk minimization in general (and SVM as a special case). Consider $\mathcal{L}_\infty(\mathcal{X})$, the space of all bounded measurable functions from $\mathcal{X} \mapsto \mathbb{R}$ and suppose we start off with a functional class $\mathcal{F} \subseteq \mathcal{L}_\infty(\mathcal{X})$[1] (or, $H \subseteq \mathcal{L}_\infty(\mathcal{X})$), where $\mathcal{X}$ is as defined in Section 3.1. Let our goal be to find a function f within $\mathcal{F}$ (or within H) that minimizes the given empirical criterion, empirical risk in ERMs (or regularized risk in SVMs). Now if the dimension d of the input space is too large, it might lead to solutions that are too complex than what is sufficient for our purpose.

[1]Note that the loss functions we consider in this paper (unless otherwise mentioned) are convex and locally Lipschitz with $\mathcal{R}_{L,P}(0) < \infty$, and hence by (2.11) and Proposition 5.27 of SC08, we have $\mathcal{R}^*_{L,P,\mathcal{L}_\infty(\mathcal{X})} = \mathcal{R}^*_{L,P}$. Hence instead of $\mathcal{L}_0(\mathcal{X})$ it suffices to consider the smaller subspace $\mathcal{L}_\infty(\mathcal{X})$.

Suppose now that the minimizer of infinite-sampled risk with respect to the oracle measure P and the functional class $\mathcal{L}_\infty(\mathcal{X})$, actually resides in $\mathcal{L}_\infty(\mathcal{X}^*)$, where $\mathcal{X}^*$ is a lower dimensional version of $\mathcal{X}$. Then it may actually suffice to find the empirical minimizer in a suitably defined lower dimensional version of $\mathcal{F}$ (or the RKHS H), and to avoid overfitting it might become a necessity. The need for defining the lower dimensional adaptations of a given arbitrary functional class (or a given RKHS) in the way of Definition 1 arises from this observation itself. Now the motivation for our algorithm stems from the heuristic belief that if some of the covariates are unimportant or superfluous for the problem at hand, the contribution of each of these variables in the functional relationship between the output variable and the covariate space in terms of the solution might be small at best, that is the incremental risk associated with a solution defined on a subset of the covariate space (by ignoring these surplus variables), when compared to the solution in the original covariate space, might indeed be minimal.

3.3.2 Further discussions on the lower dimensional spaces $\mathcal{F}^J$ (or H^J)

First note that for a given input space $\mathcal{X}$, $\mathcal{X}^J$ may not be a subspace of $\mathcal{X}$. However the assertion holds trivially for any Euclidean open ball B centered at 0. So we assume that $\mathcal{X} \subseteq B \subset \mathbb{R}^d$. We will also assume that we can sufficiently extend $\mathcal{F}(\mathcal{X})$ to $\mathcal{F}(B)$ (or, $H_\mathcal{X}$ to H_B when H is a RKHS), such that the domain of functions in $\mathcal{F}(B)$ (or in the RKHS H_B) is B instead of $\mathcal{X}$. In case of the RKHS H, this in turn extends the domain of the kernel k from $\mathcal{X} \times \mathcal{X}$ to $B \times B$. Hence from here onwards we will assume $\mathcal{X}^J \subseteq \mathcal{X}$. Note also that $\mathcal{F}^J$ may not be a subspace of $\mathcal{F}$ (that is, H^J may not be a subspace of the RKHS H). Although it is more desirable for these functional classes to accept a nested structure between each other, so that as we go down from a space to its lower dimensional version, it may not hold in general.

We now provide a few results that connect these lower dimensional spaces with the original one. In view of Definition 1, we can define $\mathcal{L}_\infty^J(\mathcal{X}) = \{f \circ \pi^{J^c} \,:\, f \in \mathcal{L}_\infty(\mathcal{X})\}$. Then Lemma 3 below says that $\mathcal{L}_\infty^J(\mathcal{X}^J) \equiv \mathcal{L}_\infty^J(\mathcal{X})\big|_{\mathcal{X}^J}$ is isomorphic to the space $\mathcal{L}_\infty(\mathcal{X}^J)$. Lemma 4 below, observes some results connecting the original RKHS with its lower dimensional versions. A related lemma, Lemma 33 is given in Appendix B.1 noting similar results for any general space. These aim to show that many of the nice properties of a given functional space are carried forward to their re-adaptations under Definition 1. We prove Lemma 3 and 33, while proof for Lemma 4 is omitted as it follows from Lemma 33 trivially. The proofs can be found in Appendix B.3.1 and B.3.2 respectively.

Lemma 3. $\mathcal{L}_\infty^J(\mathcal{X}^J) = \mathcal{L}_\infty(\mathcal{X}^J)$.

Lemma 4. *Let $H \subset \mathcal{L}_\infty(\mathcal{X})$ be a non-empty RKHS on $\mathcal{X}$. Then for any $J \subset \{1, 2, \ldots, d\}$,*

1. *If H is dense in $\mathcal{L}_\infty(\mathcal{X})$, then H^J is dense in $\mathcal{L}_\infty(\mathcal{X}^J)$.*
2. *If the $\|\cdot\|_\infty$ closure $\overline{B_H}$ of the unit ball B_H is compact, then so is $\overline{B_{H^J}}$.*
3. *If H is separable, then so is H^J.*
4. *$e_i(id : H^J \mapsto L_\infty(\mathcal{X})) \leq e_i(id : H \mapsto L_\infty(\mathcal{X}))$, where $e_i(id : H \mapsto L_\infty(\mathcal{X}))$ is the i^{th} entropy number of the unit ball B_H of the RKHS H, with respect to the $\|\cdot\|_\infty$-norm (see Appendix B.2.2 for a definition of entropy numbers).*

3.3.3 RKHS in lower dimensions

Note that in SVMs, the minimization is computed over an RKHS, and the properties of RKHSs dictate a lot of the statistical properties of SVMs. Hence, while defining these lower dimensional spaces we need to ensure that these spaces are RKHSs as well. To

that effect, we begin this section by providing an alternate way to define the lower dimensional versions of a given RKHS that preserves the reproducing property.

Definition 5. *For a given RKHS H indexed by a kernel k and a set of indices $J \subseteq \{1, 2, .., d\}$, define $H^J \equiv H_{k\circ\pi^{J^c}}(\mathcal{X})$, where $k \circ \pi^{J^c}(x, y) := k(\pi^{J^c}(x), \pi^{J^c}(y))$.*

Note immediately that Definition 5 allows us to create lower dimensional versions of an RKHS H in a way which ensures that these spaces are RKHS as well. This inevitably questions the validity of Definition 1. We however show below that both Definitions 1 and 5 actually yield the same RKHS space H^J. We begin with the following result due to Paulsen (2009).

Proposition 6. *Let $\mathcal{S}$ be any set and $\varphi : \mathcal{S} \mapsto \mathcal{X}$ be a map. Let $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ be the kernel on $\mathcal{X}$. If we define the map $k \circ \varphi : \mathcal{S} \times \mathcal{S} \mapsto \mathbb{R}$ as, $k \circ \varphi(s, t) = k(\varphi(s), \varphi(t))$, then $k \circ \varphi$ is a kernel on $\mathcal{S}$. (Paulsen 2009, Proposition 5.13).*

The next theorem then gives a natural relationship between RKHSs $H(k)$ on $\mathcal{X}$ and $H(k \circ \varphi)$ on $\mathcal{S}$. It also implies that when $\mathcal{S}$ is a subset of $\mathcal{X}$ and φ is the inclusion *id* map of $\mathcal{S}$ into $\mathcal{X}$, the kernel $k \circ \varphi$ becomes the restriction of the kernel k on $\mathcal{S} \times \mathcal{S}$.

Theorem 7. *Let $\mathcal{X}$ and $\mathcal{S}$ be two sets and let $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ be a kernel function on $\mathcal{X}$ and let $\varphi : \mathcal{S} \mapsto \mathcal{X}$ be a function. Then $H(k \circ \varphi) = \{f \circ \varphi : f \in H(k)\}$, and for $g \in H(k \circ \varphi)$ we have that $\|g\|_{H(k\circ\varphi)} = \inf\{\|f\|_{H(k)} \ : \ g = f \circ \varphi\}$.*

See Paulsen (2009) for a proof of Theorem 7.

Now let $\mathcal{X}_0$ be a subset of $\mathcal{X}$ and $k^{(0)}(x, y)$ be the restriction of a kernel k on $\mathcal{X}_0$. Let $H_k(\mathcal{X})$ be the RKHS with respect to $k(x, y)$, and $H_{k^{(0)}}(\mathcal{X})$ be the one with respect to $k^{(0)}(x, y)$. Then by the above theorem, if we define φ to be the inclusion id map from $\mathcal{X}_0$ to $\mathcal{X}$, we have $H_{k^{(0)}}(\mathcal{X}_0) = \{f|_{\mathcal{X}_0} : f \in H_k(\mathcal{X})\}$ and $\|g\|_{H_{k^{(0)}}} = \min\{\|f\|_{H_k} \ : \ f|_{\mathcal{X}_0} = g\}$ for $g \in H_{k^{(0)}}(\mathcal{X}_0)$.

Taking $\mathcal{X}_0 \equiv \mathcal{X}^J$ and $k^{(0)}(x,y) \equiv k(\pi^{J^c}(x), \pi^{J^c}(y))$, we immediately obtain our assertion.

3.3.4 Notion of risk in Lower Dimensional Versions of the Input Space

Note that the functional space $\mathcal{F}^J$ (and equivalently RKHS H^J) is defined on the entire input space $\mathcal{X}$ and not only on $\mathcal{X}^J$. So we can define risk for a function $f_J \in \mathcal{F}^J$ (or $f_J \in H^J$) for the entire input space $\mathcal{X}$ and not just for $\mathcal{X}^J$. Hence for a probability distribution P on $\mathcal{X} \times \mathcal{Y}$, define $\mathcal{R}_{L,P}(f_J)$ as $\mathcal{R}_{L,P}(f_J) = \int_{\mathcal{Y}} \int_{\mathcal{X}} L(y, x, f_J(x)) P(x,y) dx dy$. This allows us to compare the risk of functions in different lower dimensional versions of the original functional space.

3.4 RFE in nested or dense models

In this section we discuss the consistency of our feature elimination algorithm (for both ERM and SVM), when the functional space considered for the problem admits nice properties, like nestedness or denseness. We begin this section by defining these properties and citing important situations when we encounter these spaces. We then discuss our inherent assumption for existence of a null model in these frameworks, and show how that translates to the idea of variable selection through our backward elimination algorithm.

3.4.1 Nested spaces in risk minimization

Often in risk minimization, the space of functions $\mathcal{F}$ we consider for optimization will admit the nested property. To explain it mathematically, for a pair $J_1, J_2 \in \{1, 2, \ldots, d\}$ with $J_1 \subseteq J_2$, the subspaces will satisfy the condition that $\mathcal{F}^{J_2} \subseteq \mathcal{F}^{J_1}$. This in turn translates to admitting nested inequalities between risk of the minimizers in these spaces of the form $\mathcal{R}^*_{L,P,\mathcal{F}^{J_1}} \leq \mathcal{R}^*_{L,P,\mathcal{F}^{J_2}}$. One simple example of such is the linear space,

where coefficients are allowed to take values in a compact interval containing 0, that is, $\mathcal{F} = \left\{ f(x_1, \ldots, x_d) = \sum_i a_i x_i \ : \ |a_i| \leq M, \ M < \infty \right\}$.

In empirical risk minimization problems with relative flexibility on the choice of the functional space $\mathcal{F}$, we can enforce the nested property even when $\mathcal{F}$ does not satisfy the nested criterion to begin with, by considering unions of it with its lower dimensional versions. Noting that $\mathcal{F} \equiv \mathcal{F}^{\emptyset}$, we can create them as follows:

$$\widetilde{\mathcal{F}}^J = \bigcup_{J \subseteq J^* \subseteq \{1,\ldots,d\}} \mathcal{F}^{J^*}. \tag{3.5}$$

It can be seen that the properties of $\mathcal{F}$ and $\mathcal{F}^J$s with respect to Lemma 33 are carried forward in our new definitions too.

Unfortunately, in general, RKHSs need not be nested in each other. And given any RKHS H, we cannot create unions of RKHSs to use them in learning, because unions of RKHSs may not be a RKHS. The question is when can these naturally occurring RKHSs be nested within each other? We will see below that dot-product kernels actually have this property.

Lemma 8. *Dot product kernels produce nested RKHSs.*

See Appendix B.3.3 for a proof. Dot product kernels (eg: linear kernels) are often very common in formulation of a SVM problem. Other kernels might also satisfy the nested criterion. We will see through discussions in Section 3.4.3 the usefulness of the nestedness property.

3.4.2 Dense spaces in risk minimization

Another wide class of functional spaces we typically consider in risk minimizaion are dense spaces. If $\mathcal{F}$ is dense in $\mathcal{L}_\infty(\mathcal{X})$, it means that $\mathcal{F}$ represents the space of bounded functions sufficiently well, and that any function in $\mathcal{L}_\infty(\mathcal{X})$ is well approximated by

some function in $\mathcal{F}$. Many times in SVMs, the RKHS we consider for optimization will be dense in $\mathcal{L}_\infty(\mathcal{X})$. Note that all universal kernels produce RKHSs that are dense in $\mathcal{L}_\infty(\mathcal{X})$ with respect to convex, locally Lipschitz continuous losses and that all non-trivial radial kernels (eg: Gaussian RBF kernel) share this property as well (see Micchelli et al. 2006).

3.4.3 Existence of a null model

In this section we show that by starting off with the assumption of the existence of a null model, we can validate our recursive elimination algorithm if the functional space $\mathcal{F}$ (or the RKHS H) satisfy any of the above properties. What we mean by existence of a null model is that, there exists an index set J_*, such that

$$\mathcal{R}^*_{L,P,\mathcal{F}} = \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}} \tag{3.6}$$

holds.

Remark 9.

1. First, note that this is not really an assumption, since J_* can be the empty set. What we mean is that if the above condition holds for a J_*, our algorithm will be able to pick it up.

2. Note that this assumption tells us that in terms of risk, we do not lose anything at all if we consider the pair $\left(\mathcal{X}^{J_*}, \mathcal{F}^{J_*}\right)$ instead of $(\mathcal{X}, \mathcal{F})$ for the problem at hand. And as mentioned before, to avoid overfitting this indeed becomes necessary.

3. We strengthen our assumption of a null model by further claiming that no other J with $J \supset J_*$ satisfies the above property. This says that the rest of the covariates (given by the index set $J_*^c \equiv Z \setminus J_*$) in the model are all important for the learning problem, and cannot be considered for redundancy.

4. Also note that the above assumptions do not claim the uniqueness of J_*. Rather we say that for any set of covariates with the above property (3.6)*, there always exist a maximal set in terms of it.*

NESTED MODELS: Simple observation then shows that in nested spaces $\mathcal{R}^*_{L,P,F} = \mathcal{R}^*_{L,P,F^{J_*}}$ transcribes into saying that there exists a minimizer $(f_{P,\mathcal{F}})$ of infinite-sample risk in $\mathcal{F}$, which also lives in $\mathcal{F}^{J_*}$, that is, $f_{P,\mathcal{F}} \in \mathcal{F}^{J_*}$. This then trivially implies that $f_{P,\mathcal{F}} \in \mathcal{F}^J$ for any $J \subseteq J_*$, which implies that (3.6) holds for any $J \subset J_*$. Now further assume that $\mathcal{F}^{J_*}$ is the smallest such subspace admitting this relationship. Hence in nested spaces under the assumption of a null model, we expect equality of risks in the form of $\mathcal{R}^*_{L,P,\mathcal{F}} = \mathcal{R}^*_{L,P,\mathcal{F}^J} = \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}}$ whenever $J \subseteq J_*$, and then we also have that for any $J_\circ \nsubseteq J_*$, $\mathcal{R}^*_{L,P,\mathcal{F}^{J_\circ}} \geq \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}} + \epsilon_0$ for some $\epsilon_0 > 0$. This essentially substantiates the elimination of features in a backward recursive manner with a given stopping criterion.

DENSE MODELS: Now if we admit $\mathcal{F}$ to be dense in $\mathcal{L}_\infty(\mathcal{X})$, Lemma 33 tells us that $\mathcal{F}^J$ is dense in $\mathcal{L}^J_\infty(\mathcal{X})$ for any $J \in \{1, 2, \ldots, d\}$. First note that for J_1 and J_2 with $J_1 \subseteq J_2$, we trivially have a nested property of the form $\mathcal{L}^{J_2}_\infty(\mathcal{X}) \subseteq \mathcal{L}^{J_1}_\infty(\mathcal{X}) \subseteq \mathcal{L}_\infty(\mathcal{X})$. This then implies $\mathcal{R}^*_{L,P,\mathcal{F}^{J_2}} \geq \mathcal{R}^*_{L,P,\mathcal{F}^{J_1}}$. Now 'denseness' does not necessarily imply 'nestedness', but we do have the 'almost nested' property in the sense that for any $g \in \mathcal{F}^{J_2}$, and for any $\epsilon > 0$, $\exists\ f_\epsilon \in \mathcal{F}^{J_1}$ with $\|f_\epsilon - g\|_\infty \leq \epsilon$. This means that if we start off with the assumption that there exists a J_* such that $\mathcal{R}^*_{L,P,F} = \mathcal{R}^*_{L,P,F^{J_*}}$, then it implies that $\exists\ \{f_n\} \in \mathcal{F}$, such that $f_n \to f_{P,\mathcal{F}^{J_*}}$. Since the loss functions we consider are locally Lipschitz continuous, by Lemma 2.17 of SC08 we have $\mathcal{R}_{L,P}(f_n) \to \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}} = \mathcal{R}^*_{L,P,\mathcal{F}}$. This then implies that for any $J \subseteq J_*$, $\mathcal{R}^*_{L,P,\mathcal{F}^J} \geq \mathcal{R}^*_{L,P,\mathcal{F}} = \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}} \geq \mathcal{R}^*_{L,P,\mathcal{F}^J}$. Hence for every $J \subseteq J_*$, $\mathcal{R}^*_{L,P,\mathcal{F}^J} = \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}}$. Now if we further assume that $\mathcal{F}^{J_*}$ is the smallest such subspace admitting this relationship, then we again come up with the relationship that, for any $J_\circ \nsubseteq J_*$, $\mathcal{R}^*_{L,P,\mathcal{F}^{J_\circ}} \geq \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}} + \epsilon_0$ for some $\epsilon_0 > 0$. Again, the premise here allows for elimination of features in a backward recursive manner with a given

stopping criterion.

3.5 Consistency Results for RFE

The main aim of this section is to show that Algorithm 2 defined in Section 3.2.1, is consistent in finding the correct feature space in nested or dense spaces.

We now state the main result of our paper. Note that $e_i\left(id : H \mapsto L_\infty(D_\mathcal{X})\right)$ is the i^{th} entropy number for the inclusion id map of RKHS H into $L_\infty(D_\mathcal{X})$ for the input data $D_\mathcal{X} := \{X_1, \ldots, X_n\}$ (see Appendix B.2.2 for a definition of entropy numbers). We also assume condition 1 below:

Condition 1.

1. The functional space is either nested or dense.

2. There exists a J_*, such that $\mathcal{R}^*_{L,P,\mathcal{F}} = \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}}$ and that J_* is the maximal set satisfying this property.

Theorem 10. *Let P be a probability measure on $\mathcal{X} \times \mathcal{Y}$, where the input space $\mathcal{X}$ is a valid metric space. Let $L : \mathcal{X} \times \mathcal{Y} \times \mathbb{R} \mapsto [0, \infty]$ be a convex locally Lipschitz continuous loss function satisfying $L(x, y, 0) \leq 1$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Let H be the separable RKHS of a measurable kernel k on $\mathcal{X}$ with $\|k\|_\infty \leq 1$. Let, for fixed $n \geq 1$, $\exists$ constants $a \geq 1$ and $p \in (0, 1)$ such that $\mathbb{E}_{D_\mathcal{X} \sim P^n_\mathcal{X}} e_i\left(id : H \mapsto L_\infty(D_\mathcal{X})\right) \leq a i^{-\frac{1}{2p}}$, $i \geq 1$, where $\mathbb{E}_{D_\mathcal{X} \sim P^n_\mathcal{X}}$ is defined as the expectation with respect to the product measure $P^n_\mathcal{X}$ under the assumption that the input data $D_\mathcal{X} \equiv \{\mathcal{X}_1, \ldots, \mathcal{X}_n\}$ are i.i.d. copies of $\mathcal{X} \sim P_\mathcal{X}$. For a given sample size n, let $\{\lambda_n\} \in [0, 1]$ be such that $\lambda_n \to 0$ and $\lim_{n\to\infty} \lambda_n n = \infty$. We assume that there exists a $c > 0$ and a $\beta \in (0, 1]$ such that $A_2^J(\lambda) \leq c\lambda^\beta$ for any J and for all $\lambda \geq 0$ (where $A_2^J(\lambda) \equiv A_2^{H^J}(\lambda)$).*

There exists $\{\delta_n\}$ such that $\delta_n = \epsilon_0 - O(n^{-\frac{\beta}{2\beta+1}})$, for which the following statements hold:

1. *The Recursive Feature Elimination Algorithm for support vector machines, defined for* $\{\delta_n\}$ *given above, will find the correct lower dimensional subspace of the input space* ($\mathcal{X}^{J_*}$) *with probability tending to* 1.

2. *The function chosen by the algorithm achieves the best risk within the original RKHS* H *asymptotically.*

Remark 11.

1. Note as mentioned before, we do not need (3.6) *to necessarily hold for a non-trivial* J_*. *If* $\mathcal{X}$ *is full, that is, if all covariates in the model are important, then* $J_* = \{0\}$, *and our algorithm shall pick* $\mathcal{X}^{J_*^c} \equiv \mathcal{X}$.

2. Also note that the conditions $L(x, y, 0) \leq 1$, *and* $\|k\|_\infty \leq 1$ *for the kernel* k *in Theorem 10 are assumed for simplicity and might be too restrictive in some settings, but equivalent conditions like* $L(x, y, 0) \leq M$ *and* $\|k\|_\infty \leq k_{sup}$ *for constants* $M, k_{sup} > 1$ *are good enough for the proofs and will result in bounds differing from the ones derived here only up to some constants.*

We refer to Appendix B.1.2 for a version of this result in empirical risk minimization setting. The proof is postponed to Section 3.8.

3.6 Case Studies I

In this section we show the validity of our results in many practical cases of risk minimization by discussing the results in some known settings.

3.6.1 CASE STUDY 1: Feature Elimination in Linear Regression

In this case study we present our results for the simple setting of linear regression. This example shows that the consistency results achieved in this paper can be applied to many different situations ranging from simple to complex risk minimization problems

and in some cases can substantiate known techniques that are in practice in such contexts for feature elimination. Linear regression is one of the most frequently used statistical techniques for data analysis. It is also a simple example of an empirical risk minimization problem.

In a linear regression model, we assume that the functional relationship can be expressed as $y = \langle \alpha, x \rangle + b_0$, where $\langle \alpha, x \rangle$ denotes the Euclidean inner product of vectors α and x and b_0 is the bias. The prediction quality of this model can be measured by the squared-error loss function L_{LS} given as $L_{LS}(x, y, f(x)) = (f(x) - y)^2$ and our goal is to find linear weights $\widehat{\alpha}$ and $\widehat{b_0}$ for the observed data D that minimize the empirical risk. We assume that the input space $\mathcal{X} \subseteq B \subset \mathbb{R}^d$. We further assume that $\mathcal{Y} \subset \mathbb{R}$ is a closed set. The functional space $\mathcal{F}_{\text{lin}}$ is given by $\mathcal{F}_{\text{lin}} = \{f_{\alpha,b_0} : f_{\alpha,b_0}(x) = \langle \alpha, x \rangle + b_0,\ (\alpha, b_0) \in \mathbb{R}^{d+1},\ \|(\alpha, b_0)\|_\infty \leq M,\ \textit{for some } M < \infty\}$. We can now observe that the regularity conditions[2] required for the consistency for the recursive algorithm in this setting hold for this problem. The Least Squares Loss function L_{LS} is convex, and as observed in SC08, L_{LS} is locally Lipschitz continuous when $\mathcal{Y}$ is compact. Existence of M and B follows from the observations that $\mathcal{X} \subseteq B \subset \mathbb{R}^d$, $\mathcal{Y} \subset \mathbb{R}$ is a closed set, and that for some $M < \infty$, $\|(\alpha, b_0)\|_\infty \leq M$ for any function f_{α,b_0} within $\mathcal{F}_{\text{lin}}$. Compactness follows trivially since $\mathcal{F}_{\text{lin}}$ is non-empty. We assume an exponential bound on the average entropy number. Many analyses have been done on covering numbers for linear function classes (see Zhang and Bartlett 2002, Williamson 2000) and under quite general assumptions it was proved that exponential bounds can be imposed on the ϵ-entropies of such functional classes, which is actually stronger than our bound (Refer to Theorems 4 and 5 in Zhang and Bartlett (2002)).

Thus the RFE procedure presented in this paper translates in the linear regression case as a non-parametric backward selection method based on the value of the 'average

[2]Refer to Appendix B.1.2 for these regularity conditions.

sum of squares of error' or R^2/n. Indeed, the average empirical risk of the estimator $\widehat{f}(x)$ for the sample is exactly R^2/n. In a non-parametric setup, under restrictive distributional assumptions on the output vector $\mathcal{Y}$, the idea of using penalized versions of $\log R^2$ like AIC, AICc or BIC are well accepted ad-hoc methodologies for model selection (and hence feature elimination), although it is not always trivial to know which penalty should be used in a given situation, or which is best in that regard. This paper produces a theoretical basis for using the non-penalized criterion R^2/n as a tool for feature elimination in linear regression. Suppose we start with a set of covariates $\mathcal{X} = \{X_1, \ldots, X_d\}$ and let's assume without loss of generality that the covariates are pre-ordered on the basis of their importance. Then null model assumption can be interpreted as claiming the existence of an $r \in \{1, 2, \ldots, d\}$ such that the following null hypothesis is true $H_0 : \{\alpha_d = \cdots = \alpha_{r+1} = 0, \alpha_r, \ldots, \alpha_1 \neq 0\}$. So this paper establishes consistency for RFE based on the criterion R^2/n and a pre-determined stopping rule in finding the correct feature space $\mathcal{X}_0 = \{X_1, \ldots, X_r\}$ under this null hypothesis H_0.

3.6.2 CASE STUDY 2: Support Vector Machines with a Gaussian RBF Kernel

Here we provide a brief review of the application of RFE in the classic SVM premise for classification using a Gaussian RBF kernel. Assume that $\mathcal{Y} = \{1, -1\}$. We want to find a function $f : \mathcal{X} \mapsto \{1, -1\}$ such that for almost every $x \in \mathcal{X}$, $P(f(x) = Y \big| \mathcal{X} = x) \geq 1/2$. In this case, the desired function is the Bayes decision function $f^*_{L,P}$ with respect to the loss function $L_{BC}(x, y, f(x)) = 1\{y \cdot \text{sign}(f(x)) \neq 1\}$. In practice, since L_{BC} is non convex, it is usually replaced by the hinge loss function $L_{HL}(x, y, f(x)) = \max\{0, 1 - yf(x)\}$. For SVMs with a Gaussian RBF kernel, we minimize the regularized empirical criterion $\lambda\|f\|^2 + \frac{1}{n}\sum_{i=1}^{n} \max\{0, 1 - y_i f(x_i)\}$ for the observed sample $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ within the RKHS $H_\gamma(\mathcal{X})$ with the kernel

k_γ defined as $k_\gamma(x,y) = e^{-\frac{\|x-y\|_2^2}{\gamma^2}}$.

Lemma 12. *For classification using support vector machines with a Gaussian RBF kernel, the RFE defined for* $\delta = \epsilon_0 - O(n^{-\frac{\beta}{2\beta+1}})$ *where* $\beta = \frac{\beta_d \tau_d}{d\beta_d + d\tau_d + \beta_d \tau_d}$, *with* $\beta_d \in (0,\infty)$ *being the margin-noise exponent of the distribution* P *on* $\mathbb{R}^d \times \{-1,1\}$ *and* $\tau_d \in (\,0,\infty\,]$ *being the tail exponent of the marginal distribution* $P_{\mathcal{X}}$, *is consistent in finding the correct feature space*[3].

In order to prove Lemma 12, we need to verify the regularity conditions given before Theorem 10 in this setup. First note that L_{HL} is Lipschitz continuous and bounded for all 3-tuples of the form $(x,y,0)$ (see Example 2.27 in SC08). Separability of H_γ holds since an RKHS over a separable metric space having a continuous kernel is separable (Lemma 4.33 of SC08) and since $\mathcal{X} \in \mathbb{R}^d$ is separable. It is also easy to see that $|k_\gamma(x,y)| \leq 1$ is true for all $x, y \in \mathcal{X}$ and all $\gamma > 0$ and hence $\|k_\gamma\|_\infty \leq 1$.

From the proof of Proposition 17 (also see results in chapter 7 of SC08) we can see that the assumption on the bound on the average entropy of the RKHS space given before Theorem 10, can be replaced by the following:

- We assume that for fixed $n \geq 1$, $\exists$ constants $a \geq 1$ and $p \in (0,1)$ such that for any $J \subseteq \{1,2,\ldots,d\}$, $\mathbb{E}_{D_{\mathcal{X}} \sim P_{\mathcal{X}}^n} e_i \left(id : H^J \mapsto L_2(D_{\mathcal{X}}) \right) \leq a i^{-\frac{1}{2p}}, \quad i \geq 1.$

It is easily seen from the steps in (5.8) in Appendix B.3.5 that results will hold if we replace the earlier assumption with the latter. Then we see that Theorem 7.34 with Corollary 7.31 of SC08 along with the fact that $d/(d+\tau)$ is an increasing function in d, yields a bound as given here with $a := \max_{d_1 \leq d} c_{\epsilon,p} \gamma^{-\frac{(1-p)(1+\epsilon)d_1}{2p}}$ for $\gamma \leq 1$, for all $\epsilon > 0$, $d/(d+\tau) < p < 1$ and a constant $c_{\epsilon,p}$ depending only on p and a given ϵ. We however preferred to use the former in our theoretical derivations because it can be potentially weaker in many situations.

[3]For a discussion on margin-noise exponents and tail exponents of a distribution refer to Chapter 8 of SC08

The bound on the approximation error follows from results obtained in Theorem 8.18 of SC08 (see also Theorem 2.7 in Steinwart and Scovel 2007). Note that this bound is not required for consistency results, as we already have that $A_2(\lambda) \to 0$ when $\lambda \to 0$ from Lemma 5.15 of SC08. It however helps us to obtain explicit rates for the RFE and we show that it holds here which will help us to derive rates in this framework. Without going into explicit details, we can see from Theorem 8.18 of SC08 that the approximation error for a SVM using Gaussian RBF kernel of width γ on $\mathbb{R}^d$ can be bounded by a function given as

$$A_2(\lambda, d, \gamma) \leq \max\{c_{d,\tau_d}, \widetilde{c}_{d,\beta_d} c_d\} \left(\lambda^{\frac{\tau_d}{d+\tau_d}} \gamma^{-\frac{d\tau_d}{d+\tau_d}} + \gamma^{\beta_d} \right), \tag{3.7}$$

where P is a distribution on $\mathbb{R}^d \times \{-1, 1\}$ that has margin-noise exponent $\beta_d \in (0, \infty)$ and whose marginal distribution $P_{\mathcal{X}}$ has tail exponent $\tau_d \in (\, 0, \infty\,]$, $c_{d,\tau_d}, \widetilde{c}_{d,\beta_d} > 0$ are constants and c_d is the constant occurring in equation (8.10) in SC08. So for a given pair $(\lambda,\ d)$ if we choose $\gamma(\lambda, d) = \lambda^{\frac{\tau_d}{d\beta_d + d\tau_d + \beta_d \tau_d}}$ then it can be seen that $A_2(\lambda, d, \gamma(\lambda, d)) \preceq \lambda^{\frac{\beta_d \tau_d}{d\beta_d + d\tau_d + \beta_d \tau_d}}$ (where $\preceq$ denotes 'less than or equal to' up to constants). Hence the bound on the approximation error is satisfied for any J.

So for a sequence of SVM objective functions $\lambda_n \|f\|^2_{H_{\gamma(\lambda_n)}} + \frac{1}{n} \sum_{i=1}^n \max\{0, 1 - y_i f(x_i)\}$ defined for a sequence $\lambda_n^{-1} = o(n)$ with $\lambda_n \to 0$ the assumptions for the theoretical results on consistency of RFE are met, and thus Lemma 12 is proved.

3.7 Assumptions for RFE in general function spaces

In this section we discuss assumptions that are inherently needed for consistency of our algorithm under more general settings. We also discuss the necessity of these assumptions for our recursive search through appropraite examples.

3.7.1 Assumptions

Consider the setting of risk minimization (regularized or non regularized) with respect to a given functional space $\mathcal{F}$ (which are typically RKHSs in case of SVM). Our aim in this section is to provide a framework where the modified recursive feature elimination method is consistent in finding the correct lower dimensional subspace of the input space. First we note the following assumptions:

(A1). Let J be a subset of $\{1, \ldots, d\}$. Let $f_{P,\mathcal{F}^J}$ be the function that minimizes risk within the space $\mathcal{F}^J$ with respect to the measure P on $\mathcal{X} \times \mathcal{Y}$. Define $\mathcal{F}^\emptyset = \mathcal{F}$. We assume that there exists a J_*, that is, $|J_*| = d - d_0$ (where d_0 is the number of significant signals in the model) with $d_0 \geq 0$, such that it satisfies the criterion that for any pair (d_1, d_2) satisfying $d_1 \leq d_2 \leq d - d_0$, $\exists$ J_{d_1} and J_{d_2} with $J_{d_1} \subseteq J_{d_2} \subseteq J_*$ and $|J_{d_1}| = d_1$ and $|J_{d_2}| = d_2$, we have the condition that $\mathcal{R}^*_{L,P,\mathcal{F}^{J_*}} = \mathcal{R}^*_{L,P,\mathcal{F}^{J_{d_1}}} = \mathcal{R}^*_{L,P,\mathcal{F}^{J_{d_2}}}$.

Remark 13.

1. In other words, Assumption (A1) says that there exists a 'path' from the original input space $\mathcal{X}$ to the correct lower dimensional space $\mathcal{X}^{J_*}$ in the sense of equality of the minimized risk within $\mathcal{F}^J$s along this 'path'. So there exists a sequence of indices $\mathcal{J}$ from $J_{start} = \emptyset$ to $J_{end} = J_*$, where $\mathcal{J} := \big\{\{J_{start} \equiv J_1, J_2, \ldots, J_{end}\} : J_1 \subseteq J_2 \subseteq \cdots \subseteq J_{end}, |J_i| = |J_{i-1}| + 1\big\}$, such that $\mathcal{R}^*_{L,P,\mathcal{F}^J}$ is the same for all $J \in \mathcal{J}$.

2. Note that $\mathcal{J}$ may not be unique and there might be more than one path leading to $\mathcal{X}^{J_*}$.

3. Also note that J_* may not be unique in general, but any one of them would work for our purpose. So we will assume it to be unique in this paper.

(A2). Let $\mathcal{J}_1, \mathcal{J}_2, \ldots, \mathcal{J}_N$ be the exhaustive list of such paths from $\mathcal{X}$ to $\mathcal{X}^{J_*}$, and let $\widetilde{\mathcal{J}} := \bigcup_{i=1}^{N} \mathcal{J}_i$. There exists $\epsilon_0 > 0$ such that whenever $J \notin \widetilde{\mathcal{J}}$, $\mathcal{R}^*_{L,P,\mathcal{F}^J} \geq \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}} +$

ϵ_0.

Note trivially from discussions we had in Section 3.4.3, that assumptions (A1) and (A2) are satisfied for nested or dense models. Now at first glance these assumptions might look restrictive, but these do help define the premise for consistency of the resursive algorithm in any general setting. In Section 3.5 we will show how Assumptions (A1) and (A2) are sufficient for a recursive feature elimination algorithm like RFE to work (in terms of consistency). The following examples however are used to show the necessity of these assumptions in order for a well-defined recursive feature elimination algorithm to work.

3.7.2 Necessity of existence of a path in (A1)

Example 14. *Consider the empirical risk minimization framework. Let* $X = [-1, 1]^2$ *and let* $Y = 0$. *Let* $X_1 \sim \mathcal{U}$ *where* $\mathcal{U}$ *is some distribution on* $[-1, 1]$ *and* $X_2 \equiv -X_1$. *Let the functional space* $\mathcal{F}$ *be* $\{c(X_1 + X_2),\, c > 0\}$. *Let the loss function be the squared error loss, i.e.,* $L(x, y, f(x)) = (y - f(x))^2$. *By Definition 1,* $\mathcal{F}^{\{1\}} = \{cX_2,\, c > 0\}$ *and* $\mathcal{F}^{\{2\}} = \{cX_1,\, c > 0\}$ *and* $\mathcal{F}^{\{1,2\}} = \{0\}$. *We see that* $\mathcal{R}_{L,P}(f_{P,\mathcal{F}}) = \mathcal{R}_{L,P}(f_{P,\mathcal{F}^{\{1,2\}}}) = 0$ *but both* $\mathcal{R}_{L,P}(f_{P,\mathcal{F}^{\{1\}}})$ *and* $\mathcal{R}_{L,P}(f_{P,\mathcal{F}^{\{2\}}}) \neq 0$. *Hence even if the correct low-dimensional functional space may have minimized risk the same as that of the original functional space, if there does not exist a path going down to that space, the recursive algorithm will not work. Note that the minimizer of the risk belongs to* $\mathcal{F}^{\{1,2\}}$ *but there is no path from* $\mathcal{F}$ *to* $\mathcal{F}^{\{1,2\}}$, *in the sense of (A1).*

3.7.3 Necessity of Equality in (A1)

It would appear that for the algorithm to work, we don't have to necessarily work with equalities along the path and that we can relax (A1) to include inequalities as well. Suppose we redefine (A1) such that the equality of minimized risk along the path is

replaced by the inequality '$\leq$'. So now we assume that minimized risk is not necessarily constant along the path, but that it does not increase. We show below that under this modified assumption, our recursive search algorithm might fail to find the correct lower dimensional subspace of the input space.

Example 15. *Consider the empirical risk minimization framework again. Let* $Y \sim U(-1,1)$ *and* $X \subset \mathbb{R}^3$ *such that* $Y = X_3 = X_2 + 1 = X_1 - 1$. *Let* $\mathcal{F} = \{c_1X_1 + c_2X_2 + c_3X_3,\ c_1,\ c_2,\ c_3 \geq 1\}$, *and let the loss function be squared error loss. Now by definition,* $\mathcal{F}^{\{1\}} = \{c_2X_2 + c_3X_3,\ c_2,\ c_3 \geq 1\}$, $\mathcal{F}^{\{2\}} = \{c_1X_1 + c_3X_3,\ c_1,\ c_3 \geq 1\}$, $\mathcal{F}^{\{3\}} = \{c_2X_2 + c_1X_1,\ c_1,\ c_2 \geq 1\}$, $\mathcal{F}^{\{1,2\}} = \{c_3X_3,\ c_3 \geq 1\}$, $\mathcal{F}^{\{1,3\}} = \{c_2X_2,\ c_2 \geq 1\}$, $\mathcal{F}^{\{2,3\}} = \{c_1X_1,\ c_1 \geq 1\}$, *and* $\mathcal{F}^{\{1,2,3\}} = \{0\}$.

By simple calculations, we see that $\mathcal{R}^*_{L,P,\mathcal{F}} = \mathcal{R}^*_{L,P,\mathcal{F}^{\{1\}}} = \mathcal{R}^*_{L,P,\mathcal{F}^{\{2\}}} = 4/3$, $\mathcal{R}^*_{L,P,\mathcal{F}^{\{3\}}} = \mathcal{R}^*_{L,P,\mathcal{F}^{\{1,2,3\}}} = 1/3$, $\mathcal{R}^*_{L,P,\mathcal{F}^{\{1,3\}}} = \mathcal{R}^*_{L,P,\mathcal{F}^{\{2,3\}}} = 1$ *and* $\mathcal{R}^*_{L,P,\mathcal{F}^{\{1,2\}}} = 0$. *Note that the correct dimensional subspace of the input space is* $X^{\{1,2\}}$ *and there exists paths leading to this space via* $X \to X^{\{1\}} \to X^{\{1,2\}}$ *since* $\mathcal{R}^*_{L,P,\mathcal{F}} = \mathcal{R}^*_{L,P,\mathcal{F}^{\{1\}}} > \mathcal{R}^*_{L,P,\mathcal{F}^{\{1,2\}}}$ *or via* $X \to X^{\{2\}} \to X^{\{1,2\}}$ *since* $\mathcal{R}^*_{L,P,\mathcal{F}} = \mathcal{R}^*_{L,P,\mathcal{F}^{\{2\}}} > \mathcal{R}^*_{L,P,\mathcal{F}^{\{1,2\}}}$ *in the sense of Assumption (A1*). But there also exists the blind path* $X \to X^{\{3\}}$ *since* $\mathcal{R}^*_{L,P,\mathcal{F}} > \mathcal{R}^*_{L,P,\mathcal{F}^{\{3\}}}$ *which does not lead to the correct subspace. Hence the recursive search in this case may not be guaranteed to lead to the correct subspace.*

Hence equality in (A1) guarantees that the recursive search will never select an important dimension $j \in J_*$ *for redundancy because then the Assumption (A2) would be violated. Hence the equality in (A1) will ensure that we will follow a path recursively to the correct input space* $\mathcal{X}^{J_*}$.

3.8 Theoretical Results

Our main goal for this section is to prove Theorem 10 in Section 3.5. Note that it was stated under Condition 1, for nested or dense spaces. The result will continue to

hold if we replace Condition 1 by the following Condition 2.

Condition 2.

1. The functional space is general.

2. Assumptions (A1) and (A2) hold.

As seen in discussions in Section 3.4.3, note that under Condition 1, Assumptions (A1) and (A2) are satisfied trivially. Our goal here is to then prove the main result under the most general setting of Condition 2. Before that however, let us provide a few relevant results that will help us in proving this theorem.

3.8.1 Additional Results

We start off with the following lemma:

Lemma 16. *Let $(\mathcal{F}, \|\cdot\|_{\mathcal{F}})$ be a separable functional space, such that the metric $\|\cdot\|_{\mathcal{F}}$ dominates pointwise convergence. Also we assume $\sup \|f\|_{\mathcal{F}} \leq C$ for some $C < \infty$ for all $f \in \mathcal{F}$. Let L be a convex, locally Lipschitz loss function such that $L(x, y, f(x)) \leq B$ for some $B < \infty$ for all $f \in \mathcal{F}$. Also assume that for fixed $n \geq 1$, $\exists$ constants $a \geq 1$ and $p \in (0, 1)$ such that $\mathbb{E}_{D_{\mathcal{X}} \sim P_{\mathcal{X}}^n} e_i\left(\mathcal{F}, L_\infty(D_{\mathcal{X}})\right) \leq a i^{-\frac{1}{2p}}, \quad i \geq 1$. Then, we have with probability greater than or equal to $1 - e^{-\tau}$,*

$$\begin{aligned}\sup_{f \in \mathcal{F}} |\mathcal{R}_{L,P}(f) - \mathcal{R}_{L,D}(f)| &\leq 2B\sqrt{\frac{2\tau}{n}} + \frac{10B\tau}{3n} \\ &+ \quad 4 \max\left\{C_1(p) c_L(C)^p a^p B^{1-p} n^{-\frac{1}{2}}, C_2(p) c_L(C)^{\frac{2p}{1+p}} a^{\frac{2p}{1+p}} B^{\frac{1-p}{1+p}} n^{-\frac{1}{1+p}}\right\}.\end{aligned}$$

See Appendix B.3.4 for a proof. This Lemma gives us a bound for comparing the empirically obtained decision function with the omniscient oracle, having an infinite number of observations, in the case of minimizing the L-risk over $\mathcal{F}$, under the given conditions. We now assume the premise of Section 3.7.1, that is we assume (A1) and (A2) both hold. The above Lemma then helps set up the next proposition, which aims

to bound the difference in the empirical decision function in SVMs, when we move between spaces, lying in the pathway hypothesized in Assumption (A1).

Proposition 17. *Again we assume P to be a probability measure on $\mathcal{X} \times \mathcal{Y}$, and that the input space $\mathcal{X}$ is a valid metric space. We will assume $L : \mathcal{X} \times \mathcal{Y} \times \mathbb{R} \mapsto [0, \infty]$ to be convex and locally Lipschitz continuous, satisfying $L(x, y, 0) \leq 1$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Again we assume H to be the separable RKHS of a measurable kernel k on $\mathcal{X}$ with $\|k\|_\infty \leq 1$, and that for fixed $n \geq 1$, $\exists$ constants $a \geq 1$ and $p \in (0, 1)$ such that $\mathbb{E}_{D_\mathcal{X} \sim P_\mathcal{X}^n} e_i \left(id : H \mapsto L_\infty(D_\mathcal{X}) \right) \leq a i^{-\frac{1}{2p}}$, $i \geq 1$. Now for a fixed $\lambda > 0$, $\epsilon > 0$, $\tau > 0$, and $n \geq 1$, and for J_1, $J_2 \in \widetilde{\mathcal{J}}$ such that $J_1 \subseteq J_2 \subseteq J_*$, we have with probability P^n not less than $1 - 2e^{-\tau}$,*

$$\begin{aligned} &\left| \lambda \left\| f_{D,\lambda,H^{J_2}} \right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D} \left(f_{D,\lambda,H^{J_2}} \right) - \lambda \left\| f_{D,\lambda,H^{J_1}} \right\|_{H^{J_1}}^2 - \mathcal{R}_{L,D} \left(f_{D,\lambda,H^{J_1}} \right) \right| \\ &< A_2^{J_1}(\lambda) + A_2^{J_2}(\lambda) + 12B\sqrt{\frac{2\tau}{n}} + 20B\frac{\tau}{n} + 24K_2 B^{1-p} \left(\frac{a^{2p}}{\lambda^p n} \right)^{\frac{1}{2}}, \end{aligned} \tag{3.8}$$

where $A_2^{J_1}(\lambda)$ and $A_2^{J_2}(\lambda)$ are the approximation errors for the two separate RKHS classes H^{J_1} and H^{J_2}, $B := c_L(\lambda^{-1/2})\lambda^{-1/2} + 1$, and

$$K_2 := \max \left\{ B^p/4, C_1(p) c_L(\lambda^{-\frac{1}{2}})^p, C_2(p) c_L(\lambda^{-\frac{1}{2}})^{\frac{2p}{1+p}} \right\}$$

is a constant depending only on B, p and the Lipschitz constant $c_L(\lambda^{-1/2})$.

See Appendix B.3.5 for a detailed proof of Proposition 17.

Note that since $B \geq 1$ and $K_2 \geq B^p/4$, we have that if $a^{2p} > \lambda^p n$,

$$\begin{aligned} \left| \lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,D} \left(f_{D,\lambda,H^J} \right) - \mathcal{R}^*_{L,P,H^J} \right| &\leq \mathcal{R}_{L,D}(0) + \mathcal{R}_{L,P}(0) \leq 2 \\ &< 3B \leq 12K_2 B^{1-p} \left(\frac{a^{2p}}{\lambda^p n} \right)^{\frac{1}{2}}. \end{aligned} \tag{3.9}$$

Similarly, since $B \geq 1$ and $K_2 \geq B^p/4$, we have for $a^{2p} > \lambda^p n$,

$$\begin{aligned}
&\lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}^*_{L,P,H^J} \\
\leq &\lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right) + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) \\
\leq &\mathcal{R}_{L,P}(0) + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) \leq 1 + B \leq 2B \\
\leq &8K_2 B^{1-p} \left(\frac{a^{2p}}{\lambda^p n}\right)^{\frac{1}{2}}. \qquad (3.10)
\end{aligned}$$

Now note that for any J, we have

$$\begin{aligned}
&\left|\lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}^*_{L,P,H^J}\right| \\
\leq &\left|\lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}^*_{L,P,H^J}\right| + \left|\mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right)\right| \\
\leq &A_2^J(\lambda) + 2 \sup_{\|f\|_{H^J} \leq \lambda^{-1/2}} \left|\mathcal{R}_{L,P}(f) - \mathcal{R}_{L,D}(f)\right| + \sup_{\|f\|_{H^J} \leq \lambda^{-1/2}} \left|\mathcal{R}_{L,P}(f) - \mathcal{R}_{L,D}(f)\right|. \qquad (3.11)
\end{aligned}$$

Consequently we obtain the following two corollaries:

Corollary 18. *Assume the conditions of Proposition 17. For any J and all $\epsilon > 0$, $\tau > 0$, and $n \geq 1$, we have with P^n probability $> 1 - e^{-\tau}$,*

$$\begin{aligned}
&\left|\lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}^*_{L,P,H^J}\right| \\
< &A_2^J(\lambda) + 6B\sqrt{\frac{2\tau}{n}} + 10B\frac{\tau}{n} + 12K_2 B^{1-p}\left(\frac{a^{2p}}{\lambda^p n}\right)^{\frac{1}{2}},
\end{aligned}$$

where K_2 is as before. Additionally, if $J \in \widetilde{\mathcal{J}}$, we can replace $\mathcal{R}^*_{L,P,\mathcal{F}^J}$ in the above inequality by $\mathcal{R}^*_{L,P,\mathcal{F}}$.

Corollary 19. Oracle Inequality for SVM: *Assume the conditions of Proposition 17. For any J and all $\epsilon > 0$, $\tau > 0$, and $n \geq 1$, we have with P^n probability*

$> 1 - e^{-\tau}$,

$$\begin{aligned}&\lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}^*_{L,P,H^J} \\ <&A_2^J(\lambda) + 4B\sqrt{\frac{2\tau}{n}} + \frac{20B\tau}{3n} + 8K_2B^{1-p}\left(\frac{a^{2p}}{\lambda^p n}\right)^{\frac{1}{2}},\end{aligned}$$

where K_2 is as before.

Proposition 17 and Corollaries 18, 19 developed for SVM will be used to prove the following Lemma 20, that will set up the premise for proving Theorem 10.

We now provide Lemma 20, which is the last result that we need before proving Theorem 10. We will now further assume that the regularization constant λ_n converge to 0 and assume the rate for such convergence is as given in Theorem 10. To explicitly establish rates for our algorithm we also assume that the bound on the approximation error $A_2^J(\lambda)$ is as given in the aforementioned theorem.

Lemma 20. *Assume the conditions of Theorem 10. Then the following statements hold:*

i. For J_1, $J_2 \in \widetilde{\mathcal{J}}$ such that $J_1 \subseteq J_2 \subseteq J_*$, $\exists\ (\{\epsilon_n\} > 0) \to 0$ such that we have with P^n probability greater than $1 - 2e^{-\tau}$,

$$\lambda_n \left\| f_{D,\lambda_n,H^{J_2}} \right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_2}}\right) \leq \lambda_n \left\| f_{D,\lambda_n,H^{J_1}} \right\|_{H^{J_1}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_1}}\right) + \epsilon_n.$$

ii. For $J_1 \in \widetilde{\mathcal{J}}$ and $J_2 \notin \widetilde{\mathcal{J}}$ and for $J_1 \subset J_2$, $\exists\ (\{\epsilon_n\} > 0) \to 0$, such that we have with P^n probability greater than $1 - 2e^{-\tau}$,

$$\begin{aligned}\lambda_n \left\| f_{D,\lambda_n,H^{J_2}} \right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_2}}\right) \geq&\lambda_n \left\| f_{D,\lambda_n,H^{J_1}} \right\|_{H^{J_1}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_1}}\right) \\ &+ \epsilon_0 - \epsilon_n.\end{aligned}$$

iii. Oracle Property for RFE in SVM: *The infinite-sampled regularized risk for the empirical solution* f_{D,λ_n,H^J}, $\lambda_n \left\| f_{D,\lambda_n,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda_n,H^J}\right)$ *converges in measure to* $\mathcal{R}^*_{L,P,H}$ *(and hence to* $\mathcal{R}^*_{L,P}$ *if the RKHS* H *is dense in* $\mathcal{L}_\infty(\mathcal{X})$*) iff* $J \in \widetilde{\mathcal{J}}$.

The proof of Lemma 20 is given in Appendix B.3.6. We are now ready to prove Theorem 10. The proof is for any general setting and hence, we assume that assumptions (A1) and (A2) hold.

3.8.2 Proof of Theorem 10

Proof. (1) Let $\mathcal{X}^{J_*}$ be the correct input space and J_* be the correct set of dimensions to be removed with $|J_*| = d - d_0$. To prove the first part of Theorem 10, we show that, starting with the input space $\mathcal{X}$, the probability that we reach the space $\mathcal{X}^{J_*}$ is 1 asymptotically. First let us assume that there exists only one correct 'path' from $\mathcal{X}$ to $\mathcal{X}^{J_*}$. Let $\mathcal{J}^\circ$ be that correct path and $\mathcal{J}^\circ = \{J_0^\circ \equiv \{\cdot\}, J_1^\circ, \ldots, J_{d-d_0}^\circ \equiv J_*\}$.

From the proof of (i) in Appendix B.3.6, we have

$$\begin{aligned}
&\lambda_n \left\| f_{D,\lambda_n,H^{J_{i+1}^\circ}} \right\|_{H^{J_{i+1}^\circ}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_{i+1}^\circ}}\right) \\
\leq &\lambda_n \left\| f_{D,\lambda_n,H^{J_i^\circ}} \right\|_{H^{J_i^\circ}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_i^\circ}}\right) + \epsilon_n
\end{aligned}$$

with probability at least $1-2e^{-\tau}$ for $\epsilon_n = (2c+24\sqrt{2\tau}+48K_2a^{2p})n^{-\frac{\beta}{2\beta+1}} + 40\tau n^{-\frac{4\beta+1}{2(2\beta+1)}}$. Now let $J_{i+1} \neq J_{i+1}^\circ$ be any other J such that $J_i^\circ \subset J_{i+1}$ with $\|J_{i+1}\| = \|J_i^\circ\| + 1$, we have from (5.15) and (5.16) in Appendix B.3.6 that

$$\begin{aligned}
&\lambda_n \left\| f_{D,\lambda_n,H^{J_{i+1}}} \right\|_{H^{J_{i+1}}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_{i+1}}}\right) \\
> &\lambda_n \left\| f_{D,\lambda_n,H^{J_i^\circ}} \right\|_{H^{J_i^\circ}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_i^\circ}}\right) + \epsilon_0 - \epsilon_n
\end{aligned}$$

with probability at least $1 - 2e^{-\tau}$. Now if we choose $\tau = o(n^{\frac{2\beta}{2\beta+1}})$ with $\tau \to \infty$, then we see $\epsilon_n = O(n^{-\frac{\beta}{2\beta+1}})$, and hence $\delta_n \equiv \epsilon_0 - \epsilon_n$ satisfies the second inequality with the condition that $\delta_n = \epsilon_0 - O(n^{-\frac{\beta}{2\beta+1}})$ with $\delta_n \to 0$. Now since ϵ_0 is a fixed constant, $\exists N_{\epsilon_0} > 0$ such that $\forall n \geq N_{\epsilon_0}$, $2\epsilon_n \leq \epsilon_0$. Without loss of generality we assume that $n \geq N_{\epsilon_0}$. Then we have the condition that

$$\begin{aligned} &\lambda_n \left\| f_{D,\lambda_n,H^{J_{i+1}^\circ}} \right\|^2_{H^{J_{i+1}^\circ}} + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_{i+1}^\circ}} \right) \\ \leq &\lambda_n \left\| f_{D,\lambda_n,H^{J_i^\circ}} \right\|^2_{H^{J_i^\circ}} + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_i^\circ}} \right) + \delta_n \end{aligned}$$

with probability at least $1 - 2e^{-\tau}$.

For notational ease, let us define,

$$\begin{aligned} \mathcal{RR}\left(J_1, J_2\right) :=& \lambda_n \left\| f_{D,\lambda_n,H^{J_1}} \right\|^2_{H^{J_1}} + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_1}} \right) \\ &- \lambda_n \left\| f_{D,\lambda_n,H^{J_2}} \right\|^2_{H^{J_2}} - \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_2}} \right), \end{aligned}$$

and $\mathcal{RR}\left(J\right) := \lambda_n \left\| f_{D,\lambda_n,H^{J}} \right\|^2_{H^{J}} + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J}} \right) - \mathcal{R}^*_{L,P,H}$.
Then,

$$\begin{aligned} &P\left(\text{`}\textit{RFE finds the correct dimensions}\text{'}\right) \\ &\geq P\left(\text{`}\textit{RFE follows the path } \mathcal{J}^\circ \textit{ to the correct dimension space}\text{'}\right) \\ &= P\left(J_0 := J_0^\circ,\ J_1 := J_1^\circ, \ldots, J_{d-d_0} := J_{d-d_0}^\circ, J_{d-d_0+1} := \emptyset\right) \\ &= P\left(J_0 := J_0^\circ\right) P\left(J_1 := J_1^\circ \middle| J_0^\circ\right) \cdots \\ &\cdots P\left(J_{d-d_0} := J_{d-d_0}^\circ \middle| J_0^\circ, \ldots, J_{d-d_0-1}^\circ\right) P\left(J_{d-d_0+1} := \emptyset \middle| J_0^\circ, \ldots, J_{d-d_0}^\circ\right), \end{aligned}$$

where '$J_{d-d_0+1} := \emptyset$' means the algorithm stops at that step. Note that $P\left(J_0 := J_0^\circ\right) = 1$

and then observe,

$$
\begin{aligned}
& P\left(J_{i+1} := J^{\circ}_{i+1} \middle| J^{\circ}_0, \ldots, J^{\circ}_i\right) \\
=& P\left(J_{i+1} := J^{\circ}_{i+1} \middle| J^{\circ}_i\right) \quad (\because \{J^{\circ}_0, \ldots, J^{\circ}_{i-1}\} \textit{ have already been removed from the model}) \\
=& P\Big(\mathcal{RR}\left(J^{\circ}_{i+1}, J^{\circ}_i\right) \leq \delta_n \,,\, \mathcal{RR}\left(J^{\circ}_{i+1}, J^{\circ}_i\right) < \mathcal{RR}\left(J^{\bullet}_{i+1}, J^{\circ}_i\right) \;\; \forall J^{\bullet}_{i+1} \neq J^{\circ}_{i+1}\Big) \\
\geq& P\Big(\mathcal{RR}\left(J^{\circ}_{i+1}, J^{\circ}_i\right) \leq \delta_n \,,\, \delta_n < \mathcal{RR}\left(J^{\bullet}_{i+1}, J^{\circ}_i\right) \;\; \forall J^{\bullet}_{i+1} \neq J^{\circ}_{i+1}\Big) \\
\geq& 1 - P\left(\mathcal{RR}\left(J^{\circ}_{i+1}, J^{\circ}_i\right) > \delta_n\right) - \sum_{J^{\bullet}_{i+1} \neq J^{\circ}_{i+1}} P\left(\mathcal{RR}\left(J^{\bullet}_{i+1}, J^{\circ}_i\right) \leq \delta_n\right) \\
\geq& 1 - 2e^{-\tau} - 2(d-i-1)e^{-\tau} \;=\; 1 - 2(d-i)e^{-\tau}.
\end{aligned}
$$

Also see that,

$P\left(J_{d-d_0+1} := \emptyset \middle| J^{\circ}_0, \ldots, J^{\circ}_{d-d_0}\right) = P\left(\mathcal{RR}\left(J_{d-d_0+1}, J^{\circ}_{d-d_0}\right) > \delta_n \;\; \forall J_{d-d_0+1}\right) \geq 1 - 2d_0e^{-\tau}$.
Hence,
$P\left(\textit{`RFE finds the correct dimensions'}\right) \geq \prod_{i=0}^{d-d_0} \left(1 - 2(d-i)e^{-\tau}\right)$.
Now for $\tau = o(n^{\frac{2\beta}{2\beta+1}})$ with $\tau \to \infty$, $P\left(\textit{`RFE finds the correct dimensions'}\right) \to 1$ as $n \to \infty$.

Now let us prove the same assertion for the case when there is more than one correct 'path' from $\mathcal{X}$ to $\mathcal{X}^{J_*}$. Let $\mathcal{J}_1, \ldots, \mathcal{J}_N$ be an enumeration of all possible such paths. Define 'C-sets' for a J_i (where index i denotes the i^{th} cycle of RFE) as $CS(J_i) := \{J_{i+1} : J_i,\ J_{i+1} \in \mathcal{J}_k \textit{ for some } k\}$. Now,

$$
\begin{aligned}
& P\left(\textit{`RFE finds the correct dimensions'}\right) \\
\geq\ & P\left(J_0 := J^{\circ}_0,\ J_1 := J^{\circ}_1 \in CS(J^{\circ}_0), \ldots\right. \\
& \left. \ldots, J_{d-d_0} := J^{\circ}_{d-d_0} \in CS(J^{\circ}_{d-d_0-1}), J_{d-d_0+1} := \emptyset\right) \\
=\ & P\left(J_0 := J^{\circ}_0\right) P\left(J_1 := J^{\circ}_1 \in CS(J^{\circ}_0) \middle| J^{\circ}_0\right) \cdots P\left(J_{d-d_0+1} := \emptyset \middle| J^{\circ}_{d-d_0}\right).
\end{aligned}
$$

Again as before $P\left(J_0 := J_0^\circ\right) = 1$ and $P\left(J_{d-d_0+1} := \emptyset \middle| J_{d-d_0}^\circ\right) \geq 1 - 2d_0e^{-\tau}$. Now note,

$$\begin{aligned}
& P\left(J_{i+1} := J_{i+1}^\circ \in CS\left(J_i^\circ\right) \middle| J_i^\circ\right) \\
\geq\ & P\left(\mathcal{RR}\left(J_{i+1}^\circ, J_i^\circ\right) \leq \delta_n\ \forall J_{i+1}^\circ \in CS\left(J_i^\circ\right),\ \delta_n < \mathcal{RR}\left(J_{i+1}^\bullet, J_i^\circ\right)\ \forall J_{i+1}^\bullet \notin CS\left(J_i^\circ\right)\right) \\
\geq\ & 1 - \sum_{J_{i+1}^\circ \in CS\left(J_i^\circ\right)} P\left(\mathcal{RR}\left(J_{i+1}^\circ, J_i^\circ\right) > \delta_n\right) - \sum_{J_{i+1}^\bullet \notin CS\left(J_i^\circ\right)} P\left(\mathcal{RR}\left(J_{i+1}^\bullet, J_i^\circ\right) \leq \delta_n\right) \\
\geq\ & 1 - 2\left|CS\left(J_i^\circ\right)\right| e^{-\tau} - 2\left|CS\left(J_i^\circ\right)^c\right| e^{-\tau} \ =\ 1 - 2(d-i)e^{-\tau},
\end{aligned}$$

since $\left|CS\left(J_i^\circ\right)\right| + \left|CS\left(J_i^\circ\right)^c\right| = d - i$. Hence again we have that,
$P\left(\text{`RFE finds the correct dimensions'}\right) \geq \prod_{i=0}^{d-d_0}\left(1 - 2(d-i)e^{-\tau}\right)$.
Hence for $\tau = o(n^{\frac{2\beta}{2\beta+1}})$ with $\tau \to \infty$, $P\left(\text{`RFE finds the correct dimensions'}\right) \to\ 1$ as $n \to \infty$.

(2) To prove the second part of Theorem 10 just observe that if J_{end} is the last cycle of the algorithm in RFE, then from (5.19) in Appendix B.3.6, and recycling arguments given at the beginning of the first part of the proof we have that

$$\begin{aligned}
& P\left(\left|\mathcal{RR}\left(J_{\text{end}}\right)\right| \leq \delta_n\right) \\
=& P\left(\left|\mathcal{RR}\left(J_*\right)\right| \leq \delta_n\right) P\left(J_{\text{end}} = J_*\right) + P\left(\left|\mathcal{RR}\left(J_{\text{end}}\right)\right| \leq \delta_n \middle| J_{\text{end}} \neq J_*\right) P\left(J_{\text{end}} \neq J_*\right) \\
\geq& P\left(\left|\mathcal{RR}\left(J_*\right)\right| \leq \delta_n\right) P\left(J_{\text{end}} = J_*\right) \\
\geq& (1 - e^{-\tau}) \prod_{i=0}^{d_0}\left(1 - 2(d-i)e^{-\tau}\right).
\end{aligned}$$

So for $\tau = o(n^{\frac{2\beta}{2\beta+1}})$ with $\tau \to \infty$,
$P\left(\left|\lambda_n \left\|f_{D,\lambda_n,H^{J_{\text{end}}}}\right\|^2_{H^{J_{\text{end}}}} + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_{\text{end}}}}\right) - \mathcal{R}^*_{L,P,H}\right| \leq \delta_n\right) \to\ 1$ with $n \to \infty$. $\square$

Note: Although (5.19) in Appendix B.3.6 was asserted for η_n, we do have $\eta_n < \epsilon_n < \delta_n\ \forall n \geq N_{\epsilon_0}$, so the proof for the second part of the theorem will hold true for δ_n.

3.9 Case Studies II

Here we further study the usage of two very important kernels in classification using support vector machines and we discuss the usefulness of our algorithm in such settings.

3.9.1 CASE STUDY 3: Protein classification with Mismatch String Kernels

A very fundamental problem in computational biology these days is the classification of proteins into functional and structural classes based on homology of protein sequence data. A new class of kernels, called the mismatch string kernels, are increasingly being used with support vector machines (SVMs) in a discriminative approach to the protein classification problem. These kernels measure sequence similarity based on shared occurrences of k length subsequences, counted with up to m mismatches. This is again a typical classification problem, where $\mathcal{Y} = \{1, -1\}$ and the hinge loss function $L_{HL}(x, y, f(x)) = \max\{0, 1 - yf(x)\}$ is again used as the surrogate loss.

The (k, m) mismatch kernel (see Leslie et al. 2004, for details) is based on a feature map from the space of all finite sequences from an alphabet $\mathcal{A}$ with $\mathcal{C}(A) = l$ to $\mathbb{Z}_{\geq 0}{}^{l^k}$, where l^k denotes the dimensions spanned by the set of k-length subsequences ('k-mers') from $\mathcal{A}$. For a fixed k-mer $\alpha = a_1 a_2 \dots a_k$, with each a_i a character in $\mathcal{A}$, the (k, m)-neighborhood generated by α is the set of all k-length sequences β from $\mathcal{A}$ that differ from α by at most m mismatches. We call this set $N_{(k,m)}(\alpha)$.

The feature map $\Phi_{(k,m)}$ for a k-mer α is defined as $\Phi_{(k,m)}(\alpha) = (\phi_\beta(\alpha))_{\beta \in \mathcal{A}^k}$, where $\phi_\beta(\cdot)$ is a indicator function such that, $\phi_\beta(\alpha) = 1$ if $\beta \in N_{(k,m)}(\alpha)$, and 0 otherwise. Then for a sequence x of any length, the feature map $\Phi_{k,m}$ is defined as follows:

$$\Phi_{(k,m)}(x) = \sum_{k-\text{mers } \alpha \text{ in } x} \Phi_{(k,m)}(\alpha),$$

that is, we extend the feature map additively by summing the feature vectors for all the k-mers in x. The (k, m)-mismatch kernel $K_{(k,m)}(x, y)$ is then the Euclidean inner product in feature space of feature vectors:

$$K_{(k,m)}(x, y) = \left\langle \Phi_{(k,m)}(x), \Phi_{(k,m)}(y) \right\rangle$$

For $m = 0$, we retrieve the k-spectral kernel. The kernel can be further normalized as,

$$K^{\text{norm}}_{(k,m)}(x, y) = \frac{K_{(k,m)}(x, y)}{\sqrt{K_{(k,m)}(x, x)}\sqrt{K_{(k,m)}(x, y)}}.$$

Feature selection in the context of protein classification is conducted on the k-mers obtained from a protein sequence instead of the original one (see Leslie et al. 2004, Iqbal et al. 2014). It is obvious that the RKHS H produced by the string kernel is finite dimensional, and hence, the coordinates of the transformed space (the k-mers) can be used directly for feature selection. Hence the problem reduces down to feature selection in linear SVMs (produced by the Euclidean inner product), and the applicability of recursive feature selection becomes clear in context of the discussions we had in Case Studies 3.6.1 and 3.6.2.

3.9.2 CASE STUDY 4: Image classification with χ^2 kernel

Indexing or retrieving images is one of the main challenges in pattern recognition problems. Using color histograms as an image representation technique is useful because of the reasonable performance that can be obtained in spite of their extreme simplicity (see Swain and Ballard 1992). Image classification using their histogram representation has become an popular option in many such settings. The support vector machine (SVM) approach is considered a good classification technique in this setting because of its high generalization performance without any prior model assumption, even when

the dimension of the input space is very high (see Chapelle et al. 1999).

Selecting the kernel is important as in any classification method with SVMs, and generalized RBF kernels of the form $K_\rho^{d-\mathrm{RBF}}(x,y) = e^{-\rho d(x,y)}$ are have been proven to be useful for classification in this context. In the case of images as input, the L_2 norm that generates the Gaussian RBF kernel seems to be quite meaningful here. However, as histograms are discrete densities, other suitable comparison functions exist, especially the χ^2 function, which has been used extensively for histogram comparisons (Schiele and Crowley 1996). The χ^2 distance is given as $d_{\chi^2}(x,y) = \sum_i \frac{(x_i-y_i)^2}{x_i+y_i}$, and hence the χ^2 kernel has the form,

$$K_\rho^{\chi^2-\mathrm{RBF}}(x,y) = e^{-\rho \sum_i \frac{(x_i-y_i)^2}{x_i+y_i}}.$$

In order to establish the consistency of our algorithm in this setting, we would need to verify the regularity conditions given before Theorem 10 in this setup. Look from the discussions in Case Study 3.6.2, we already established the conditions of the Hinge loss function L_{HL}. The input space $\mathcal{X}$ in image classification are histograms which can be represented as $h \times w$ vectors for grayscale images and $3 \times h \times w$ vectors, where h and w are the height and width of the images in pixels. It is easy to see that the kernel $K_\rho^{\chi^2-\mathrm{RBF}}$ is continuous, and the input space is separable, hence separability of $H_\rho^{\chi^2-\mathrm{RBF}}$ follows from Lemma 4.33 of SC08. It is also easy to see that $\|K_\rho^{\chi^2-\mathrm{RBF}}\|_\infty \leq 1$.

Note that the input space $\mathcal{X}$ can be included in a Euclidean ball and the kernel $K_\rho^{\chi^2-\mathrm{RBF}}$ is infinitely many times differentiable. Then by Theorem 6.26 of SC08, we have explicit polynomial bounds on the i^{th} entropy number of RKHS generated by these kernels in essence of the assumption given in Theorem 10. Also note that the polynomial bound we assume on the approximation error $A_2(\lambda)$ helps us to obtain explicit rates for the RFE, but there hasn't been much work done on the theoretical derivations of properties of support vector machine classification with a χ^2 kernel, and

this still remains an open problem in this domain. However consistency follows in spite of any such assumption on the approximation error as $A_2(\lambda) \to 0$ when $\lambda \to 0$ from Lemma 5.15 of SC08. The above discussions validate RFE as a useful technique for feature selection in image classification problems.

3.10 Simulation Study

In this section we present a short simulation study to illustrate the use of risk-RFE for feature elimination in SVMs and compare it with penalized methods, like LASSO.

3.10.1 Consistency and selection of features

Note that the use of RFE for feature selection has been in practice for well over a decade and it is a well-accepted technique in classification. The main aim of this section is to evaluate our consistency results, and a method for selection of the subset of features. We consider two different data-generating mechanisms, one in the classical classification setting and the other in regression. For each of these examples we again look at three different scenarios. For the first scenario, the total number of covariates is 15 of which only 4 are important. For the second scenario, there are 30 covariates with only 7 important ones. The third scenario has 50 covariates with 3 that are important.

For the classification example we consider the hinge loss L_{HL} as the surrogate loss and the SVM function is computed using the Gaussian RBF kernel $k_\gamma(x_1, x_2) = \exp\{-\frac{1}{\gamma^2}\|x_1 - x_2\|_2^2\}$. The covariates X were generated uniformly on the segment $[-1,\ 1]$ and the output vector Y was generated as $Y = \text{sign}(\omega' X)$, where ω is the vector of coefficients with the first few elements non-zero, corresponding to the important features, chosen at random from a list of coefficients $[-1, -0.5, 0.5, 1]$ and the rest are zero. We initialize the original SVM function using a 5-fold cross validation on the kernel width

SVM-RBF	$d = 15$, $d_0 = 4$			$d = 30$, $d_0 = 7$			$d = 50$, $d_0 = 3$		
(vs LASSO)	n=100	n=200	n=400	n=100	n=200	n=400	n=100	n=200	n=400
Prop no errors (a)	0.97 (0.94)	1 (1)	1 (1)	0.62 (0.46)	1 (0.96)	1 (1)	0.95 (0.97)	1 (1)	1 (1)
Prop 1 error (b)	0.03 (0.06)	0 (0)	0 (0)	0.34 (0.49)	0 (0.04)	0 (0)	0.05 (0.93)	0 (0)	0 (0)
Prop > 1 error (c)	0 (0)	0 (0)	0 (0)	0.04 (0.05)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
SVR-Linear	$d = 15$, $d_0 = 4$			$d = 30$, $d_0 = 7$			$d = 50$, $d_0 = 3$		
(vs LASSO)	n=100	n=200	n=400	n=100	n=200	n=400	n=100	n=200	n=400
Prop no errors (a)	1 (0.93)	1 (1)	1 (1)	1 (0.47)	1 (0.91)	1 (1)	1 (0.98)	1 (1)	1 (1)
Prop 1 error (b)	0 (0.07)	0 (0)	0 (0)	0 (0.48)	0 (0.09)	0 (0)	0 (0.02)	0 (0)	0 (0)
Prop > 1 error (c)	0 (0)	0 (0)	0 (0)	0 (0.05)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)

Table 3.1: Accuracy of RFE (vs LASSO)

γ and the regularization parameter λ and they were chosen from the set of values

$$\left(\frac{2}{n\lambda}, \gamma\right) = (0.01 \times 10^i,\ j) \qquad i = \{0, 1, 2, 3, 4\},\ j = \{1, 2, 3, 4\} \tag{3.12}$$

where n is the sample size for the given setting.

In the second case we used an SVR function with a linear kernel $k(x_1, x_2) = \langle x_1, x_2 \rangle$ to treat the regression setting. The loss function we considered is the ϵ-insensitive Loss $L_\epsilon(x, y, f(x)) = \max\{0, |y - f(x)| - \epsilon\}$ with $\epsilon = 0.1$. Covariates are generated as before while Y is now generated as $Y = \omega' X + \frac{1}{3} N_{\dim(X)}(0, 1)$. As before we initialize with a 5-fold Cross Validation on λ.

We repeat the process for different sample sizes $n = \{100, 200, 400\}$. We also repeat the simulations 100 times each to note down the proportion of times the RFE made no errors (a), made only one error (b) or made more than 1 error (c) (See Table 3.1), where a mistake is made if the rank of any non-important feature is found to be higher than that of any important one. We compare the performance of RFE with LASSO in both settings (logistic regression with LASSO or linear regression with LASSO depending on the example), and the results for LASSO are given in the parentheses. The simulated relationships between the output Y and the input X being linear, we should

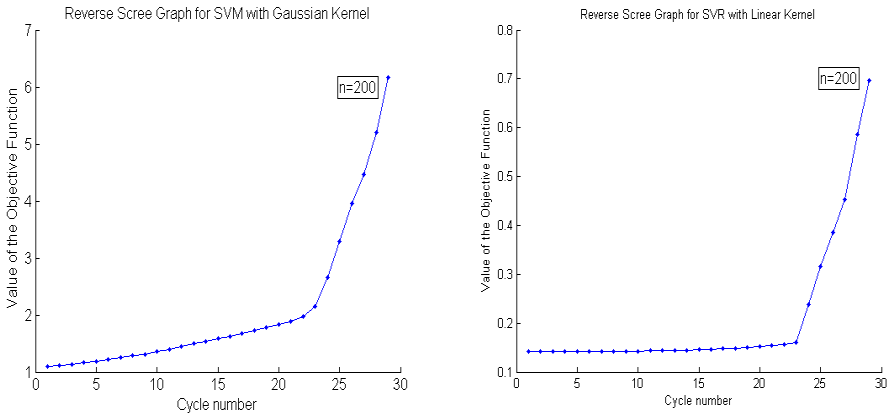


Figure 3.1: Reverse Scree Graph for one run of the simulations for (a) SVM with Gaussian Kernel (b) SVR with Linear Kernel with $d = 30$, $d_0 = 7$

expect LASSO to work as well in these settings. However as seen in Table 3.1, RFE dominates LASSO in smaller sample sizes, while in larger sample sizes both perform equally well. The entire methodology was implemented in the MATLAB environment. For the implementation we used the SPIDER library for MATLAB[4], which already has a feature elimination algorithm based on RFE and we modified it accordingly to suit our criterion for reduction. The codes for the algorithm and the simulations are given in 3.13.

One important question we inevitably face in feature elimination is when to stop. Note that our theoretical results suggest the existence of a gap ϵ_0 and our results show that asymptotically the difference in the empirical versions of the objective functions exceed it whenever we move beyond the correct dimension. Practically it is almost impossible to characterize this gap for a given setting, but the existence of this gap can be observed from the values of the objective function at each stage of the algorithm. One idea that can be implemented is that of a 'reverse Scree graph' (See section on Scree graphs in chapter 6 of Jolliffe (2002)). Implementation of the Scree graph is a

[4]The Spider library for Matlab can be downloaded from `http://www.kyb.tuebingen.mpg.de/bs/people/spider/`

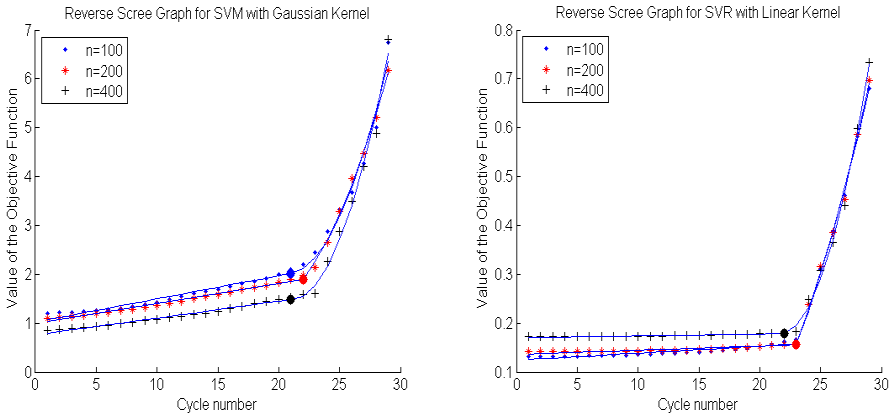


Figure 3.2: Linear-Quadratic mixture change point analysis for (a) SVM with Gaussian Kernel for comparable cross validation values of λ and kernel width γ and (b) SVR with Linear Kernel for comparable cross validation values of λ, with $d = 30$, $d_0 = 7$ for varying sample sizes. The bold dots represent the estimated change points.

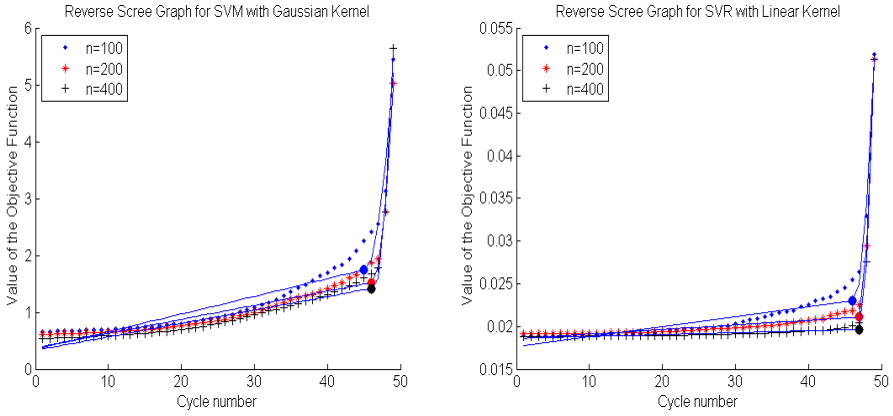


Figure 3.3: Linear-Quadratic mixture change point analysis for (a) SVM with Gaussian Kernel for comparable cross validation values of λ and kernel width γ and (b) SVR with Linear Kernel for comparable cross validation values of λ, with $d = 50$, $d_0 = 3$ for varying sample sizes. The bold dots represent the estimated change points.

well-formulated idea in choosing the correct number of Principal Components in PCA and that same idea can be applied here as well. We plot the values of the objective function $\inf_{i \in Z \setminus J} \lambda \left\| f_{D,\lambda,H^{J\cup\{i\}}} \right\|^2_{H^{J\cup\{i\}}} + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J\cup\{i\}}}\right)$ at each run of the algorithm in a graph. Figure 3.1 justifies such an argument.

For a further exploratory analysis of this gap and to characterize the number of

features to be eliminated, we tried some ad-hoc model diagnostic tools. From a heuristic standpoint, the phenomenon captured in Figure 3.1 seems to suggest that if we fit a regression model to the observed objective function values in the scree plot, we will expect a change in the slope of the regression line right after we start eliminating significant covariates because of the aforementioned gap. One plausible way to analyze this gap is to fit a change point regression model of the observed values on the number of cycles of RFE and to infer that the estimated change point is the ad-hoc stopping rule, so as to eliminate all features ranked below that point. For the asymptotic belief that the change in the objective function is negligible to the left of the change point, we fit a linear trend there. However to the right of the change point, these changes might show non-linear trends, and hence we tried linear and quadratic trends to model that. The quadratic trend seemed to work better. Some plots (see Figures 3.2, 3.3) are given here to show our analysis where we show the mixture of linear-quadratic fits.

So heuristically it is possible to justify the choice of the correct dimensions (features) based on a reverse scree graph. Otherwise some other user-defined choices for the gap size can be used to determine how many features are required in a specific setting.

3.10.2 RFE vs penalized methods

In this section, we look at some non-linear settings to establish the generability of RFE vs l_1 penalized methods. As we mentioned before, l_p penalized methods fail to find the correct subset of features in general non-linear relationships as these following simulation examples will hope to prove.

We again consider two settings: classification and regression. The covariates X is generated uniformly from the $[-2, 2]^{10}$ in both settings. In the classification example, the output variable Y depends only on table the first two features as the following: Y takes the value 1 inside the smaller square ($-1 \leq X_1 \leq 1, -1 \leq X_2 \leq 1$), and takes the

Method	Test Misclassification Error	
	Mean	Standard Error
SVM with RFE	0.051	0.0264
SVM without RFE	0.242	0.0476
Logistic Regression with Lasso	0.286	0.0734
L_1 SVM	0.262	0.0567

Table 3.2: SVM-wRFE v SVM-woRFE v Lasso v l_1 SVM

Method	Test Measurement Error	
	Mean	Standard Error
SVR with RFE	15.523	0.5885
SVR without RFE	16.293	0.0448
Linear Regression with Lasso	17.754	1.3258

Table 3.3: SVR-wRFE v SVR-woRFE v Lasso

value -1 inside the annulus formed between this smaller square and the larger square given as $(-2 \leq X_1 \leq 2, -2 \leq X_2 \leq 2)$. In the regression setting, Y again depends only on the first two features, defined by the functional relationship $Y = \frac{a_1 X_1 X_2}{(1+a_2 X_1)^2}$, where a_1, a_2 are strictly positive constants. In classification, we compare RFE (with Gaussian RBF kernel and hinge loss) with logistic regression with LASSO and L1-SVM, and in regression, we compare RFE (with Gaussian RBF kernel and ϵ-insensitive loss) with linear regression with LASSO. The results are given in the tables below. RFE was able to pick the first two features for the model satisfactorily, while the penalized methods struggled to find the same. The misclassification error in the classification setting and the measurement error in tables 3.2 and 3.3 respectively shows that RFE performs much better than these penalized methods.

3.11 High dimensional framework when p grows with n

Most of our results in the body of the draft assume the premise that we have a fixed design at hand, that is, we assume that dimension d of the input data $\mathcal{X}$ remains fixed. We derived our asymptotic results for consistency of the feature selection algorithm under this premise. High dimensional settings (when d grows with n) are becoming more and more vogue in supervised learning problems and hence, one interesting question is then to study the properties of our algorithm when both $n, d \to \infty$ (however we still assume that the number of significant signals in the design remain fixed, that is, d_0 is fixed and finite). In this section, our goal is to discuss our algorithm in light of this new premise, and modify arguments to achieve consistency like in fixed design settings.

Let us assume that $\mathcal{X} \in \mathbb{R}^d$, and we observe data $D = \{(X_1, Y_1), \ldots, (X_n, Y_n)\} \sim$ i.i.d. $P^d_{\mathcal{X}\times\mathcal{Y}}$, where the probability distribution of the design now depends on the dimension d of the input space $\mathcal{X}$. Note that P^d denotes the measure for the initial input-output space $\mathcal{X} \times \mathcal{Y}$, and as we traverse down in the feature space for our algorithm, we will assume that the probability measure on the reduced input spaces are just restrictions of P^d on these spaces (like we do for a fixed design). Henceforth, we will denote the problem by P^d. The modified feature selection algorithm is given below.

Algorithm 21. *Start off with $J \equiv [\cdot]$ empty and let $Z \equiv [1, 2, ..., d]$.*

STEP 1: In the k^{th} cycle of the algorithm choose dimension i_k for which

$$i_k = \underset{i \in Z \setminus J}{\arg\min}\, \lambda \left\| f_{D,\lambda,H^{J\cup\{i\}}} \right\|^2_{H^{J\cup\{i\}}} + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J\cup\{i\}}}\right) - \lambda \left\| f_{D,\lambda,H^J} \right\|^2_{H^J} - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right).$$

STEP 2: Update $J = J \cup \{i_k\}$. Go to STEP 1.

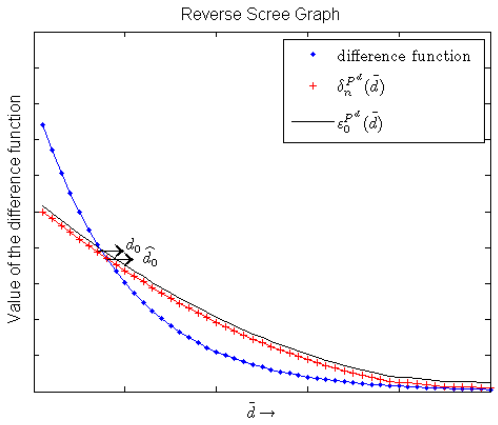


Figure 3.4: Stopping rule for the modified algorithm in the limiting design size setting

Continue this until the difference

$$\min_{i \in Z \setminus J} \lambda \left\| f_{D,\lambda,H^{J\cup\{i\}}} \right\|^2_{H^{J\cup\{i\}}} + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J\cup\{i\}}}\right) - \lambda \left\| f_{D,\lambda,H^{J}} \right\|^2_{H^{J}} - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J}}\right)$$
$$> \delta_n^{P^d}\left(d - |J|\right),$$

where $\delta_n^{P^d}(\cdot)$ is a known positive function intrinsic to the design, and output J as the set of indices for the features to be removed from the model.

So the main modification of the algorithm lies in the stopping rule. In the fixed design problem, the stopping rule was a fixed constant δ_n, while in this modified version it is a function $\delta_n^{P^d}(\cdot) : \{1, \ldots, d\} \mapsto \mathbb{R}$. Figure 3.4 shows a visual representation of the stopping condition in this case. $\delta_n^{P^d}(\cdot)$ acts as an envelop function and our algorithm is stopped if and when the difference function jumps above $\delta_n^{P^d}(\cdot)$.

To achieve consistency for this algorithm, we will now have to modify our assumptions and we will briefly discuss these modifications here. Let us consider the most general framework (Condition 2). We keep assumption (A1) fixed, that is, while moving down between spaces that always contain all the significant features, we still believe in the existence of a path of equality of risk like before. Assumption (A2) needs to be

modified however, since the assumption of a fixed gap ϵ_0 between risks in models that contain all significant features vs all other sub-optimal models makes sense only in a fixed design problem. In a varying design problem, heuristically this gap should diminish as well and shrink to 0 as d tends to ∞. Hence assumption (A2) is modified to (A2*) and is given below:

(A2*). Let $\mathcal{J}_1, \mathcal{J}_2, \ldots, \mathcal{J}_N$ be the exhaustive list of such paths from $\mathcal{X}$ to $\mathcal{X}^{J_*}$, and let $\widetilde{\mathcal{J}} := \bigcup_{i=1}^{N} \mathcal{J}_i$. There exists a monotonically decreasing discrete function $\epsilon_0^{P^d}(\cdot) > 0$ intrinsic to the problem and reaching 0 in limit, such that for $J_1 \in \widetilde{\mathcal{J}}$, $J_2 \notin \widetilde{\mathcal{J}}$ with $|J_2| = |J_1| + 1$, we have

$$\mathcal{R}^*_{L,P^d,\mathcal{F}^{J_2}} \geq \mathcal{R}^*_{L,P^d,\mathcal{F}^{J_1}} + \epsilon_0^{P^d}\left(d - |J_1|\right). \tag{3.13}$$

So we modify our assumption to reflect the varying gap size with the size of the design. Heuristically what this gap-size assumption says is the following: For a problem P^d, with starting design size d, $\epsilon_0^{P^d}(\cdot)$ is a strictly positive, monotonically decreasing function from $\{1, \ldots, d\} \mapsto \mathbb{R}$, such that $\epsilon_0^{P^d}(\tilde{d}) \to 0$ in limit when both $d, \tilde{d} \to \infty$. Hence there are two different asymptotic conditions working on $\delta_n^{P^d}(\cdot)$ here, with $\delta_n^{P^d}(\cdot) \to \epsilon_0^{P^d}(\cdot)$ as $n \to \infty$, and additionally $\delta_n^{P^d}(\tilde{d}) \to 0$ as $d, \tilde{d}, n \to \infty$.

3.11.1 Under universal bounds for entropy and approximation error

We still have some work left before we can argue consistency for this algorithm. For now, we assume that regularity conditions given in Theorem 10 will hold for any given design d, that is, there are universal constants a, c such that the entropy bound and the approximation error bound continue to hold universally. Then in lieu of our discussions in this section, simple observation will show that results stated in Lemma 16 – Corollary 19 continue to hold under slightly restated versions (P^n is replaced with $P^{d,n}$ to denote

the appropriate probability measure for the starting design). Statements (i) and (iii) in Lemma 20 will continue to hold, while (ii) can be changed to the following:

ii*. For $J_1 \in \widetilde{\mathcal{J}}$ and $J_2 \notin \widetilde{\mathcal{J}}$ and for $|J_2| = |J_1| + 1$, $\exists\ (\{\epsilon_n\} > 0) \to 0$, such that we have with $P^{d,n}$ probability greater than $1 - 2e^{-\tau}$,

$$\lambda_n \left\| f_{D,\lambda_n,H^{J_2}} \right\|^2_{H^{J_2}} + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_2}}\right) \geq \lambda_n \left\| f_{D,\lambda_n,H^{J_1}} \right\|^2_{H^{J_1}} + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_1}}\right) + \epsilon_0^{P^d}\left(d - |J_1|\right) - \epsilon_n.$$

Under the premise of this modified statement, we can sufficiently move on to establish the consistency arguments. It can be easily observed that the initial steps in the proof of Theorem 10 in section 3.8.2 (which has been presented for a fixed design size) continue to hold by taking $\delta_n^{P^d}\left(d - |J|\right) = \epsilon_0^{P^d}\left(d - |J|\right) - \epsilon_n$ for design $\mathcal{X}^J$, and now we further assume that $\sup_{d\in\mathbb{N},\tilde{d}\leq d} \liminf_{n\to\infty} \frac{\epsilon_0^{P^d}(\tilde{d})}{\epsilon_n} > 2$. This allows us to define a sequence $\{N_1, \ldots, N_d, \ldots\}$, such that $2\epsilon_n \leq \epsilon_0^{P^d}(\tilde{d})$, whenever $n > N_d$ and for all $\tilde{d} \leq d$. The subsequent steps follow and see that we arrive at,

$$\begin{aligned} P\left(\text{`RFE finds the correct dimensions'}\right) &\geq \prod_{i=0}^{d-d_0} \left(1 - 2(d-i)e^{-\tau}\right) \\ &\gtrsim \left(1 - 2de^{-\tau}\right)^d, \end{aligned}$$

where the last approximate inequality follows assuming $2de^{-\tau} < 1$ for sufficiently large n, and $\tau = o(n^{\frac{2\beta}{2\beta+1}})$ with $\tau \to \infty$. Now for the limiting infinite product to converge to 1 when $n, d \to \infty$, see that

$$\left(1 - 2de^{-\tau}\right)^d = \left(\left(1 - \frac{2d}{e^\tau}\right)^{-\frac{e^\tau}{2d}}\right)^{-\frac{2d^2}{e^\tau}}.$$

Hence if we assume $d^2 e^{-\tau} \to 0$, see that the above quantity converge to 1 in limit. Hence for consistency results to hold, d needs to grow slower than a certain rate in terms of the sample size n. See that restricting the growth of τ to be $o(n^{\frac{2\beta}{2\beta+1}})$ implies that we can choose $\tau \approx n^{\frac{2\beta k}{2\beta+1}}$ for some $k < 1$. This implies that $de^{-\tau/2} \approx de^{-0.5n^{\frac{2\beta k}{2\beta+1}}}$, and hence $d \approx o\left(e^{0.5n^{\frac{2\beta k}{2\beta+1}}}\right)$ suffices.

3.11.2 Under relaxed bounds for entropy and approximation error

It can be well reasoned that the entropy bounds (and the approximation error bounds) should depend on the size of the design d. A look at the bounds derived for the Gaussian RBF kernel in section 3.6.2 makes it clear. It is however difficult to obtain explicit bounds in terms of the design size and is currently beyond the scope of this discussion. We will then assume very relaxed rates for these bounds in terms of the design size, and try to establish our consistency arguments under that premise. Let us restate our main theorem now.

Theorem 22. *Let P^d be a probability measure on $\mathcal{X} \times \mathcal{Y}$, where the input space $\mathcal{X}$ is a valid metric space. Let $L : \mathcal{X} \times \mathcal{Y} \times \mathbb{R} \mapsto [0, \infty]$ be a convex locally Lipschitz continuous loss function satisfying $L(x, y, 0) \leq 1$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Let H be the separable RKHS of a measurable kernel k on $\mathcal{X}$ with $\|k\|_\infty \leq 1$. Let, for fixed $n \geq 1$, $\exists$ constants $\tilde{a} \geq 1$, $\alpha \geq 0$ and $p \in (0, 1)$ such that $\mathbb{E}_{D_\mathcal{X} \sim P_\mathcal{X}^{d,n}} e_i \left(id : H \mapsto L_\infty(D_\mathcal{X})\right) \leq ae^{\alpha d} i^{-\frac{1}{2p}}$, $i \geq 1$. For a given sample size n, let $\{\lambda_n\} \in [0, 1]$ be such that $\lambda_n \to 0$ and $\lim_{n\to\infty} \lambda_n n = \infty$. We also assume that there exists a $c > 0$, $\tilde{\alpha}$ and a $\beta \in (0, 1]$ such that $A_2^J(\lambda) \leq \tilde{c} e^{\tilde{\alpha} d} \lambda^\beta$ for any J and for all $\lambda \geq 0$ (where $A_2^J(\lambda) \equiv A_2^{H^J}(\lambda)$).*

For $d = O(\log n)$, there exists $\delta_n^{P^d}(\cdot) = \epsilon_0^{P^d} - O(n^{-\gamma})$ where $\gamma \in \left(0, \frac{\beta}{2\beta+1}\right)$, for which the following statements hold:

1. *The Recursive Feature Elimination Algorithm for support vector machines, defined for $\delta_n^{P^d}(\cdot)$ given above, will find the correct lower dimensional subspace of the input*

space ($\mathcal{X}^{J_*}$) with probability tending to 1.

2. *The function chosen by the algorithm achieves the best risk within the original RKHS H asymptotically.*

Now it is then well understood that the modifications needed to reflect these changes is look at our bounds in Lemma 16 – Corollary 19 by replacing a by $\tilde{a}e^{\alpha d}$ and c by $\tilde{c}e^{\tilde{\alpha}d}$. Lemma 20 can now be restated by replacing ϵ_n by $\epsilon_{n,d} = (2ce^{\tilde{\alpha}d} + 24\sqrt{2\tau} + 48K_2a^{2p}e^{2\alpha pd})n^{-\frac{\beta}{2\beta+1}} + 40\tau n^{-\frac{4\beta+1}{2(2\beta+1)}}$. Now we need to ensure that asymptotically $\epsilon_{n,d}$ goes to 0. Observe first for $\tau = o(n^{\frac{2\beta}{2\beta+1}})$, this reduces to $\epsilon_{n,d} = \widetilde{K}e^{\overline{\alpha}d}n^{-\frac{\beta}{2\beta+1}} + o(1)$, for a constant $\widetilde{K}$ and $\overline{\alpha} = \max(\tilde{\alpha}, 2\alpha p)$. Now if we fix a constant $\gamma \in \left(0, \frac{\beta}{2\beta+1}\right)$, such that $\epsilon_{n,d_n} = O(n^{-\gamma})$, we must have $e^{\overline{\alpha}d} \leq C_1 n^{\frac{\beta}{2\beta+1}-\gamma}$, or that $d = O(\log n)$. All subsequent steps follow similarly as discussed in the previous section, where we continue to assume $\sup_{d\in\mathbb{N},\tilde{d}\leq d} \liminf_{n\to\infty} \frac{\epsilon_0^{P^d}(\tilde{d})}{\epsilon_{n,d}} > 2$.

Now since $\log n$ grows slower than $e^{0.5n^{\frac{2\beta k}{2\beta+1}}}$, we have $de^{-\tau} \to 0$ for $d = O(\log n)$ automatically, and hence we can arrive at our consistency results.

3.12 Concluding remarks

We proposed an algorithm for feature elimination in empirical risk minimization and support vector machines. We studied the theoretical properties of the method, discussed the necessary assumptions, and showed that it is universally consistent in finding the correct feature space under these assumptions. We provided case studies of a few of the many different scenarios where this method can be used. Finally, we give a short simulation study to illustrate the method and discuss a practical method for choosing the correct subset of features.

Note that Lemma 20(ii) establishes the existence of a gap in the rate of change of the objective function at the point where our feature elimination method begins removing

essential features of the learning problem. This motivated us to use a scree plot of the values of the objective function at each cycle, and indeed our simulation results support our approach by visually exhibiting this gap. Moreover, the graphical interpretation of the scree plot motivated the use of change point regression to select the correct feature space. It would be interesting to conduct a more detailed and formal analysis of this gap in real life settings to facilitate more efficient, automated practical solutions.

As far as our knowledge goes, not much analysis have been done on the properties of variable selection algorithms under such general assumptions on the probability generating mechanisms of the input space, especially in support vector machines. So the results generated in this paper can act as a good starting point for similar analyses in other settings. It would also be interesting to analyze RFE for other settings, including censored support vector regression (See Goldberg and Kosorok (2013)) or other machine learning problems, including reinforcement learning or other penalized risk minimization problems.

3.13 Supplementary Materials

Details on the codes are given in the html page
`http://www.bios.unc.edu/~kosorok/RFE.html`.

CHAPTER 4: FEATURE SELECTION IN Q LEARNING

4.1 Reinforcement Learning: Methods and concepts

Let us briefly discuss the history and development of reinforcement learning (and Q learning) in the context of dynamic treatment regime.

4.1.1 Reinforcement Learning

Reinforcement learning is a computational approach to understanding and automating goal-directed learning and decision-making, and is distinguished from other approaches by its emphasis on learning from the direct interaction between an individual and its environment. A detailed account of the history of reinforcement learning is given in Sutton and Barto (1998).

In a typical reinforcement learning design, we consider a multistage decision problem with say T decision points. Let S_t be the (random) state of the patient at stage $t \in \{1, \ldots, T+1\}$ and let $\mathbf{S}_t = \{S_1, \ldots, S_t\}$ be the vector of all states up to and including stage t. Similarly, let A_t be the action chosen in stage t, and let $\mathbf{A}_t = \{A_1, \ldots, A_t\}$ be the vector of all actions up to and including stage t. Lower case letters, such as s and a, are used to denote the realizations of the random variables S and A, respectively. Hence we have $\mathbf{s}_t = \{s_0, s_1, \ldots, s_t\}$, and $\mathbf{a}_t = \{a_0, a_1, \ldots, a_t\}$. We assume that the finite longitudinal trajectories are sampled at random from a distribution P and we denote the expectation by E.

After each time step t, the patient receives a reward R_t for the treatment he/she receives, denoted possibly as a random function of the state variables up to the current

state $\mathbf{S}_t$, the actions taken each stage up to the current state $\mathbf{A}_t$, and the resultant next state S_{t+1}. When $t = 0, 1, \ldots, T$, the reward is given by $R_t = r(\mathbf{S}_t, \mathbf{A}_t, S_{t+1})$, where r is the time-dependent deterministic function specifying the relationship between the reward and the state and action variables.

In reinforcement learning, at each stage t our goal is to choose a_t so as to maximize or minimize the expected discounted return:

$$\tilde{R}_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \cdots + \gamma^T R_{t+T} = \sum_{k=0}^{T-t} \gamma^k R_{t+k},$$

where γ is the discount rate ($0 \leq \gamma \leq 1$) and controls the balance between a patient's immediate reward and future rewards.

Another important aspect of a reinforcement learning process is the exploration policy or a probability assignment p, which is defined as a map $(\mathbf{s}_t, \mathbf{a}_{t-1}) \mapsto p_t(a|\mathbf{s}_t, \mathbf{a}_{t-1})$. The policy can possibly be a deterministic action as well, that is $\pi_t(\mathbf{s}_t, \mathbf{a}_{t-1}) = a_t$. The entire sequential policy, or the sequence of deterministic decision rules $\{\pi_1, \ldots, \pi_T\}$ is called a dynamic treatment regime. Let P_π be the distribution, from which the training data are sampled, when the policy π is used to generate actions. Based on the conditional history $(\mathbf{s}_t, \mathbf{a}_{t-1})$ before the start of treatment at time t, we formulate a value function to account for the total reward a patient is expected to achieve over the future:

$$V_t(\mathbf{s}_t, \mathbf{a}_{t-1}) = E_\pi \left[\sum_{k=0}^{T-t} \gamma^k R_{t+k} | \mathbf{S}_t = \mathbf{s}_t, \mathbf{A}_{t-1} = \mathbf{a}_{t-1} \right].$$

Then the optimal value function can be defined as

$$V_t^*(\mathbf{s}_t, \mathbf{a}_{t-1}) = \max_{\pi \in \Pi} V_t(\mathbf{s}_t, \mathbf{a}_{t-1}) = \max_{\pi \in \Pi} E_\pi \left[\sum_{k=0}^{T-t} \gamma^k R_{t+k} | \mathbf{S}_t = \mathbf{s}_t, \mathbf{A}_{t-1} = \mathbf{a}_{t-1} \right], \quad (4.1)$$

where Π denote the collection of all policies. The main goal of any reinforcement

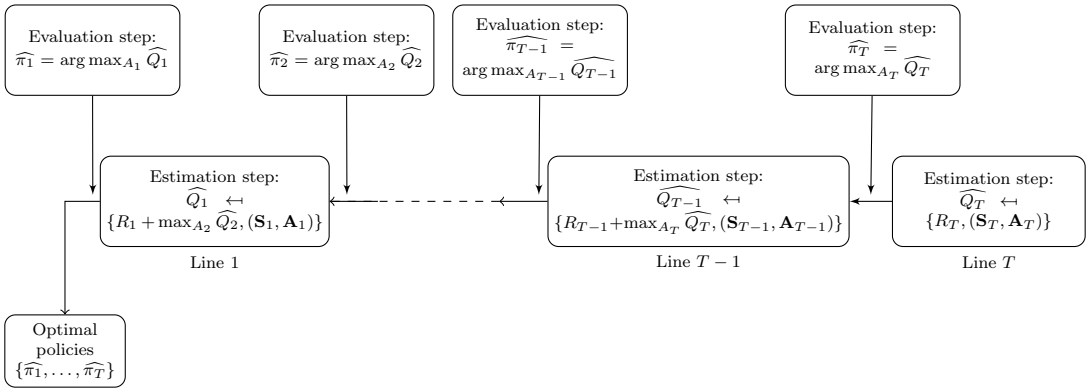


Figure 4.1: Steps of Q Learning

learning algorithm is to estimate the optimal value function efficiently. The Bellman equation (Bellman 1956) characterizes the optimal policy π^* as one that satisfies the following recursive relation:

$$\pi_t^*(\mathbf{s}_t, \mathbf{a}_{t-1}) \in \arg\max_{a_t} E\left[R_t + \gamma V_{t+1}^*(\mathbf{S}_{t+1}, \mathbf{A}_t) | \mathbf{S}_t = \mathbf{s}_t, \mathbf{A}_t = \mathbf{a}_t\right]. \tag{4.2}$$

The main goal of reinforcement learning is to find a policy that leads to a high expected cumulative reward. Naively, one could learn the transition distribution functions and the reward function using the observed trajectories, and then solve the Bellman equation recursively. However, this approach is inefficient both computationally and memory-wise. In the following section, we introduce the Q-learning algorithm, which requires less memory and less computation.

4.1.2 Q Learning

One of the most important algorithms to solve the reinforcement learning problem is Watkins' Q-learning (Watkins 1989, Watkins and Dayan 1992). Q-learning uses backward recursion to compute the Bellman equation without the need to know the full dynamics of the process. Hence, Q-learning does not estimate the value function directly, it however estimates a Q-function instead. More formally, we define the optimal

time-dependent Q-function as:

$$Q_t^*(\mathbf{s}_t, \mathbf{a}_t) = E[R_t + \gamma V_{t+1}^*(\mathbf{S}_{t+1}, \mathbf{A}_t)|\mathbf{S}_t = \mathbf{s}_t, \mathbf{A}_t = \mathbf{a}_t].$$

Now, since $V_t^*(\mathbf{s}_t, \mathbf{a}_{t-1}) = \max_{a_t} Q_t^*(\mathbf{s}_t, \mathbf{a}_{t-1}, a_t)$, it is then relatively easy to see that an optimal policy will satisfy $\pi_t^*(\mathbf{s}_t, \mathbf{a}_{t-1}) = \arg\max_{a_t} Q_t^*(\mathbf{s}_t, \mathbf{a}_{t-1}, a_t)$. Then the one-step Q-learning has the simple recursive form

$$Q_t^*(\mathbf{s}_t, \mathbf{a}_t) = E[R_t + \gamma \max_{a_{t+1}} Q_{t+1}^*(\mathbf{S}_{t+1}, \mathbf{A}_t, a_{t+1})|\mathbf{S}_t = \mathbf{s}_t, \mathbf{A}_t = \mathbf{a}_t]. \tag{4.3}$$

The recursive form of Q-learning above allows the Q_t's to be estimated backwards through time $t = T, T-1, \ldots, 1, 0$. For convenience, Q_{T+1} is set equal to 0 and the estimate beginning at the last time point $\hat{Q}_T$ is estimated and the rest are estimated recursively using the estimates from the later time points, back to $\hat{Q}_0$ at the beginning. Once the backwards estimation process is done, the sequence of $\{\hat{Q}_0, \hat{Q}_1, \ldots, \hat{Q}_T\}$ can be used for estimating optimal policies

$$\hat{\pi}_t(\mathbf{s}_t, \mathbf{a}_{t-1}) = \arg\max_{a_t} \hat{Q}_t(\mathbf{s}_t, \mathbf{a}_{t-1}, a_t)$$

where $t = 0, 1, \ldots, T$, and these optimal policies can therefore be used to test or predict for a new data set. Unless otherwise mentioned, we would assume $\gamma = 1$.

4.2 Recursive Feature Elimination

With recent development in the ease of collection and handling of large amounts of data, more often than not we have huge information at our disposal, especially with respect to the number of explanatory variables or 'features'. The incremental information provided by each of these features may often be redundant, and learning

the functional connection between the explanatory variables and the response from such high-dimensional data can be quite challenging. One way to overcome this problem is to use feature elimination techniques to find a smaller set of features that is able to perform the learning task sufficiently well.

Recursive Feature Elimination as a technique to rank features and select the optimal subset of features for learning in support vector machines (SVM) was first formulated by Guyon et al. (2002). They developed this as a backward elimination procedure based on recursive computations of the SVM learning function. At each recursive step of the algorithm, the change in the RKHS norm of the estimated SVM function is calculated after deletion of each of the features remaining in the model, and then removing the one that shows the lowest change in such norm, thus performing an implicit ranking of features. RFE and other methods derived out of RFE are generalizable in the sense that they can work in learning in a variety of complex functional classes (not just the linear space as do most of the embedded methods for feature learning in SVMs). However, arguments for RFE have mostly been heuristic, and their ability to produce successful data-driven performances have been examined only in simulated or observed data. Theoretical properties of it has never been studied in rigorous detail.

To create a method in the spirit of the generability achieved by Guyon et al. but with concrete theoretical properties, we developed a modified RFE procedure in Dasgupta et al. (2013), using a different criterion for deletion and ranking of features to enable theoretical consistency. The ranking of the features are done based on the lowest difference observed in the regularized empirical risk after removing each of those features from the existing model. The heuristic reasoning behind this is that if any of the features do not contribute to the model at all, the increase in the regularized risk will be inconsequential. This allows RFE to be generalized to the much broader yet simpler setting of empirical risk minimization where we can apply the same idea to empirical

risk.

4.2.1 The support vector machine algorithm

Let H be an $\mathbb{R}$-Hilbert space over the input space $\mathcal{X}$. A function $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is called a *reproducing kernel* of H if $k(\cdot, x) \in H$ for all $x \in \mathcal{X}$, and has the reproducing property $f(x) = \langle f, k(\cdot, x) \rangle$ for all $f \in H$, and all $c \in \mathcal{X}$. The space is called a real-valued *Reproducing Kernel Hilbert Space (RKHS)* over $\mathcal{X}$.

Let H be a separable RKHS of a measurable kernel k on $\mathcal{X}$, and fix a $\lambda > 0$. Let L be a convex and locally Lipschitz continuous loss function. Then the *empirical SVM decision function* can be defined as,

$$f_{D,\lambda,H} = \arg\min_{f \in H} \lambda \|f\|_H^2 + \mathcal{R}_{L,D}(f), \tag{4.4}$$

where D is the data $D := \{(X_1, Y_1), \ldots, (X_n, Y_n)\} \in (\mathcal{X} \times \mathcal{Y})^n$, and $\mathcal{R}_{L,D}(f)$ is the empirical risk of the function f in estimating the output variable $\mathcal{Y}$.

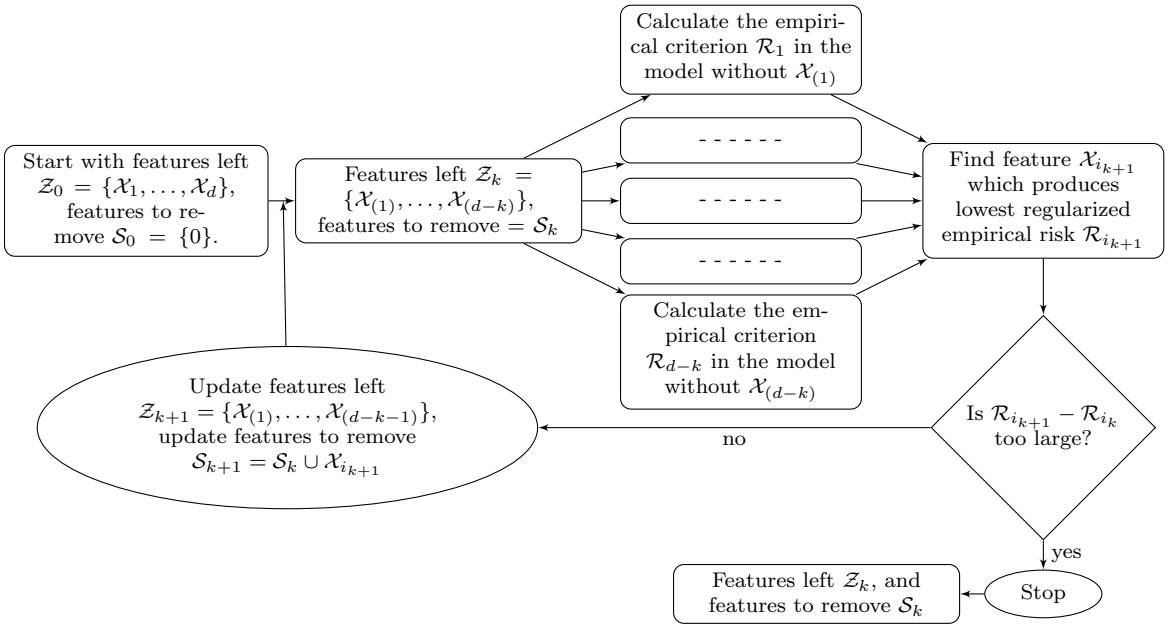


Figure 4.2: Schematics of RFE in nonparametric estimation

The infinite sampled version of the regularized minimizer is given as $f_{P,\lambda,H} = \arg\min_{f \in H} \lambda \|f\|_H^2 + \mathcal{R}_{L,P}(f)$, where P is the underlying probability distribution of the product space $(\mathcal{X} \times \mathcal{Y})$.

4.2.2 Feature Elimination Algorithm

We began by proposing a way such that starting off with an arbitrary space $\mathcal{F}$, we are able to create lower dimensional versions of it. This is indeed necessary, since at each stage of the feature elimination process, we move down to a 'lower dimensional' feature space and the functional spaces need to be adjusted to cater to the appropriate version of the problem in these subspaces.

Definition 23. *For any set of indices $J \subseteq \{1, 2, .., d\}$ and a given functional space $\mathcal{F}$, define $\mathcal{F}^J = \{g : g = f \circ \pi^{J^c} \quad , \forall f \in \mathcal{F}\}$, where π^{J^c} is the projection map from $x \mapsto x^J$ $(x, x^J \in \mathbb{R}^d)$, such that x^J is produced from x by replacing those elements in x which are indexed in the set J, by zero.*

We can hence define the space $\mathcal{X}^J = \{\pi^{J^c}(x) \, : \, x \in \mathcal{X}\}$, such that $\pi^{J^c} : \mathcal{X} \mapsto \mathcal{X}^J$ is a surjection. Now we are ready to provide the algorithm. Assume the support vector machine framework, where we are given an RKHS H indexed by a kernel k.

Algorithm 24. *Start off with $J \equiv [\cdot]$ empty and let $Z \equiv [1, 2, ..., d]$.*

1. *In the k^{th} cycle of the algorithm choose dimension i_k for which*

$$i_k = \arg\min_{i \in Z \setminus J} \lambda \left\| f_{D,\lambda,H^{J \cup \{i\}}} \right\|_{H^{J \cup \{i\}}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J \cup \{i\}}}\right) \qquad (4.5)$$
$$- \lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right).$$

2. *Update $J = J \cup \{i_k\}$. Go to STEP 1.*

Continue this until the difference

$$\min_{i \in Z \setminus J} \lambda \left\| f_{D,\lambda,H^{J\cup\{i\}}} \right\|^2_{H^{J\cup\{i\}}} + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J\cup\{i\}}}\right) - \lambda \left\| f_{D,\lambda,H^{J}} \right\|^2_{H^{J}} - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J}}\right)$$

becomes larger than a pre-determined quantity δ_n, and output J as the set of indices for the features to be removed from the model.

4.3 Feature elimination in Q learning

Before going further, let us give a more detailed account of the mechanisms of Q-learning. Typically Q_ts are modeled as a function of a set of parameters θ, where the estimator are allowed to have different parameter sets for different time points t. For example, $Q_t(\mathbf{s}_t, \mathbf{a}_t)$ may be of the form

$$Q_t(\mathbf{s}_t, \mathbf{a}_t; \theta_t) = \sum_{j=1}^{k} \theta_{tj} \phi_{tj}(\mathbf{s}_t, \mathbf{a}_t)$$

where $\theta_t = (\theta_{t1}, \ldots, \theta_{tk})$ and $\{\phi_{t1}, \ldots, \phi_{tk}\}$ are selected basis functions (See Zhao et al. 2009). The estimated optimal policies $\hat{\pi}_t(\mathbf{s}_t, \mathbf{a}_{t-1}) = \arg\max_{a_t} \hat{Q}_t(s_t, a_t; \theta_t)$, $t = 0, 1, \ldots, T$, can therefore be used to test or predict for a new data set.

Our next aim is to estimate the Q-function for finding the optimal policy. However, that is often challenging for the structure of the true Q-functions may be complex, the maximization in equation (4.3) may be non-smooth, or the state and the action spaces may be high-dimensional. A number of different approaches have been employed to obtain the estimator of interest in recent years. Murphy (2005b), Blatt et al. (2004) and Tsitsiklis and Van Roy (1996) showed that Q-learning estimation can be viewed as approximate least-squares value iteration. The parameter estimators $\hat{\theta}_t$ for the t^{th}

Q-function satisfy

$$\hat{\theta}_t \in \arg\min_{\theta} \mathbb{E}_n \left[R_t + \max_{a_{t+1}} \hat{Q}_{t+1}(\mathbf{S}_{t+1}, \mathbf{A}_t, a_{t+1}; \hat{\theta}_{t+1}) - Q_t(\mathbf{S}_t, \mathbf{A}_t; \theta_t) \right]^2, \tag{4.6}$$

where $\mathbb{E}_n$ is the empirical expectation. This is precisely the one-step update of Sutton and Barto (1998) when $\gamma = 1$. In Murphy et al. (2006), Q-learning was modeled as a generalization of the familiar regression model. Linear regression methods can work well, but for that the dimension of the action space needs to be small. Otherwise nonparametric or semi-parametric regression become desirable for estimating the Q-functions. In Zhao et al. (2009) the authors considered two flexible techniques from the machine learning literature, Support Vector Regression (SVR) and Extremely Randomized Trees (ERT), as methods to fit Q-functions and to learn an optimal policy using a training data set.

As is true with all other learning methods, reinforcement learning can suffer from 'Curse of Dimensionality'. Although support vector regression is a penalized risk minimization method and does allow for some control on the over-complexification of the estimated Q-functions, it is still necessary for some form of feature selection procedure to effectively control for overfitting and redundancy. The foremost aim of the reinforcement learning procedure is maximization of the value function, which is equivalent to minimizing risk in a related framework. It is obvious then that each stage Q-function estimation is basically a risk minimization problem. In our future research, we want to find out if the recursive feature elimination procedure that we introduced in the first part of this dissertation might be an interesting idea in this scenario, and that remains the main question of interest. Since the Q-function estimation is done recursively in a multistage format, one interesting question is whether we can tailor the RFE procedure effectively to cater to this multistage risk minimization procedure, i.e., utilize RFE on the entire multistage format to eliminate features that are surplus to the problem and

redundant in this regard.

4.4 Methods for feature selection in Q learning

In Zhao et al. (2009), they explored the use of support vector regression (SVR) and extremely randomized trees (ERT), as methods to fit Q-functions and to learn an optimal policy using a training data set. The popularity of support vector machines (SVM) as as a set of supervised learning algorithms is motivated by the fact these methods are easy-to-compute techniques that enable estimation under weak or no assumptions on the distribution (see Steinwart and Chirstmann 2008). Although the results given in Section 3 cater to the framework of support vector machines, we showed in Dasgupta et al. (2013) that RFE can be implemented in estimation methods involving empirical risk minimization as well. Hence an interesting idea in terms of feature selection in Q learning with support vector machines or other non parametric methods of estimation, can be to use RFE at each stage of estimation of the Q functions.

4.4.1 Recursive feature elimination on the estimation steps

Recursive feature elimination (RFE) discussed in section 4.2, is a technique for feature elimination in various risk minimization problems. In Dasgupta et al. (2013) we explicitly established results for consistency of the algorithm in choosing the right subset of features in support vector machines or empirical risk minimization problems (and hence in randomized trees), and potentially it can be adapted to other estimation scenarios as well. In Q learning (See Figure 4.1), we sequentially estimate the Q-functions backwards in time. At each stage of estimation t, we fit a function non parametrically to characterize the relationship between the current history H_t (where $H_t = (\mathbf{S}_t, \mathbf{A}_{t-1})$), current treatment A_t and the pseudo response $R_t + \max_{a_t} \hat{Q}_{t+1}(\mathbf{S}_{t+1}, \mathbf{A}_t, a_{t+1})$. Hence at each stage of estimation, we can use RFE to eliminate a subset of features (given

by the index set $\widehat{J}_t^*$), such that the updated history $H_t^{\widehat{J}_t^*}$ contains only those features that have been deemed important by our algorithm. And then the estimation of the Q-function Q_t is then conducted on the updated history $H_t^{\widehat{J}_t^*}$ instead of initial history H_t.

Now we give our first algorithm. It uses RFE on each estimation stage of the Q learning algorithm.

Algorithm 25 (RFE)**.** *Assume* $\hat{Q}_{T+1} = 0$. *Let us denote* $\boldsymbol{H}_t = (\boldsymbol{S}_t, \boldsymbol{A}_{t-1})$. *For* $t = T, T-1, \ldots, 1$,

1. *Use Algorithm 24 to obtain subset* $\boldsymbol{H}_t^{\widehat{J}_t^*} \subseteq \boldsymbol{H}_t$.
2. *Estimate* $\hat{Q}_t$ *based on the updated history* $(\boldsymbol{H}_t^{\widehat{J}_t^*}, A_t)$ *and pseudo response* $R_t + \max_{a_{t+1}} \hat{Q}_{t+1}(\boldsymbol{h}_{t+1}^{\widehat{J}_{t+1}^*}, a_{t+1})$.
3. *Obtain* $\hat{\pi}_t = \arg\max_{a_t} \hat{Q}_t(\boldsymbol{h}_t^{\widehat{J}_t^*}, a_t)$.

The version of the algorithm we proposed in Dasgupta et al. (2013) utilizes the trained risk (trained regularized risk in support vector machines) as the criterion for elimination. At each stage of the algorithm, we calculate the risk (or regularized risk) in the trained submodels created by removing the remaining features in the model, one at a time. The submodel that achieves the minimum risk (or regularized risk) among them is chosen and becomes the new model for the next stage of the algorithm. Thus sequentially it ranks features, and with a valid stopping rule, it can perform feature selection and select the correct subset of features. Our theoretical results suggest the existence of a gap that separates submodels having the correct subset of features as a subset, from the submodels that do not contain all the necessary features. In Dasgupta et al. (2013) we used a change point regression model to select this subset of features. The trained regularized risk achieved from the chosen submodel at each stage of the algorithm is plotted in a graph, and then a change point model is fit to estimate the

cycle of the algorithm where the graph of the objective function changes slope, and all features with lower ranks than the feature removed at that stage of the algorithm are removed for redundancy.

4.4.2 Recursive feature elimination on estimation steps using separate data folds for model training and testing

The next algorithm we propose for feature selection in Q learning is similar in essence to the one proposed earlier, but differs in the criterion for deletion. Algorithm 24 proposes and algorithm 25 utilizes the version of RFE that uses regularized risk obtained from the trained model as a rule to eliminate features. The new algorithm we propose uses separate models for training the data and testing the error rate. Before running the feature selection algorithm, we divide the observed data $D = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ into two random splits D_{train} and D_{test}. We fit our submodels in the same way as before using the training data D_{train}, but obtain estimates of the measurement error (or risk) from the test data D_{test}. Hence in support vector machines say, the regularized risk is created by adding the RKHS norm of the estimated function times the regularization parameter to the risk estimate from the test data, and is used as the criterion for deletion. This modified RFE (we call it the test-RFE) algorithm is given below:

Algorithm 26. *Start off with $J_0 \equiv [\cdot]$ empty and let $Z \equiv [1, 2, ..., d]$. Divide the observed data $D = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ into two random splits D_{train} and D_{test}, the training and the test data respectively.*

In the k^{th} cycle of the algorithm,

1. *Choose dimension i_k for which*

$$i_k = \underset{i \in Z \setminus J_{k-1}}{\arg\min}\, \lambda \left\| f_{D_{train}, \lambda, H^{J_{k-1} \cup \{i\}}} \right\|^2_{H^{J_{k-1} \cup \{i\}}} + \mathcal{R}_{L, D_{test}} \left(f_{D_{train}, \lambda, H^{J_{k-1} \cup \{i\}}} \right). \quad (4.7)$$

2. *Update* $J_k = J_{k-1} \cup \{i_k\}$.

3. *If* $|Z \setminus J_k| > 1$, *go to STEP* 1.

Find k_{stop} *such that,*

$$k_{stop} = \arg\min_k \lambda \left\| f_{D_{train},\lambda,H^{J_k}} \right\|^2_{H^{J_k}} + \mathcal{R}_{L,D_{test}} \left(f_{D_{train},\lambda,H^{J_k}} \right)$$

and output $J_{k_{stop}}$ *as the set of indices for the features to be removed from the model.*

The intuitive idea behind the rationale to replace the training set risk estimate with the test set risk estimate in the objective function is to further minimize the possibility of overfitting of the data to affect the elimination procedure. For moderate sample sizes, when the input dimension of the feature space is high compared to the number of signals in the model, it is likely that for the observed data, the model might overfit itself within the noisy dimensions satisfactorily. In that case using the training data to calculate the risk estimates in the initial steps of the algorithm might inflate the risk of elimination of the relatively weaker signals, while random variations in the data might be misclassified as important patterns. To safeguard against this possibility, we utilize the test data to calculate the risk associated with the fitted submodels, and use this as a surrogate for the training risk in the objective function. Heuristically, the test data would be free of any signature of the random patterns that contributes to overfitting in the training set, and hence perhaps gives a more coherent importance of the deleted features through the risk estimates. And as we go down in the feature space, the chances of overfitting diminishes in steps as we reach the correct dimension of features, and hence the estimate of risk in the test data is expected to decrease all the way down to this subspace of the input space. However, as we go down further below the subspace representing the significant features in the model, the risk estimate in the test set is expected to increase again, displaying a sharp bend in the objective

function right after we cross this boundary. This actually allows more ease in selection of the stopping rule, which can be the cycle of the algorithm where the sharp bend, or the minima is observed.

In Q learning then, RFE_test can be used similarly to eliminate features at each estimation phase. We present the second algorithm below:

Algorithm 27 (RFE_test)**.** *Assume* $\hat{Q}_{T+1} = 0$*. Let us denote* $\boldsymbol{H}_t = (\boldsymbol{S}_t, \boldsymbol{A}_{t-1})$*. For* $t = T, T-1, \ldots, 1$*,*

1. *Use Algorithm 26 to obtain subset* $\boldsymbol{H}_t^{\widehat{J}_t^*} \subseteq \boldsymbol{H}_t$*.*
2. *Estimate* $\hat{Q}_t$ *based on the updated history* $(\boldsymbol{H}_t^{\widehat{J}_t^*}, A_t)$ *and pseudo response* $R_t + \max_{a_{t+1}} \hat{Q}_{t+1}(\boldsymbol{h}_{t+1}^{\widehat{J}_{t+1}^*}, a_{t+1})$*.*
3. *Obtain* $\hat{\pi}_t = \arg\max_{a_t} \hat{Q}_t(\boldsymbol{h}_t^{\widehat{J}_t^*}, a_t)$*.*

4.4.3 Recursive feature elimination on the final maximization step

The final goal of the Q learning algorithm is to estimate the optimal treatment sequence $\{\hat{\pi}_{1,i}, \ldots, \hat{\pi}_{T,i}\}$ for individual i. The optimal value function at time $t = 1$ is given as $V_1^*(S_1) \equiv V^*(S_1)$, and for individual i with baseline history $s_{1,i}$, it can be calculated from the estimated Q-function $\hat{Q}_1$ as $\hat{V}(s_{1,i}) = \max_{a_1} \hat{Q}_1(s_{1,i}, a_1)$. Our goal is then to find the optimal trajectory of treatment rules for each individual based on his/her history, such that the sumtotal of rewards the individual receives at the end of the trial achieves the highest among all such possible treatment trajectories. Hence the optimal reward defines a complex relationship within the history variables $S_1 \times (S_2, A_1) \times \cdots \times (S_{T+1}, A_T)$. Heuristically we can extend the theoretical justification of our feature elimination algorithm from the framework of function estimation by minimizing a criterion function to the ultracomplex sequential network of stepwise

function estimation and maximization of the estimated function along a given direction in Q learning.

If the relationship within the history space defined by the optimal reward function can be meaningfully expressed by a lower dimensional subspace of the space spanned by the history variables, then we can meaningfully get rid of the rest and concentrate on this lower dimensional subspace for the entire algorithm. Or in other words, if there are redundant variables in the history space (then so called noise), then the information loss would not be significant for the purpose of finding the optimal regimes, if we can somehow shrink the estimation space to represent only these essential variables. This would significantly decrease the chances of overfitting, and hence could potentially result in improved optimal rewards. So intuitively if we solve the problem in this reduced space we would expect the estimated Q functions at each stage of estimation to mimic the Q functions estimated from the entire history space, and resultantly the estimated optimal rewards for both these problems should be similar.

Suppose $\mathbf{H}_t = (\mathbf{S}_t, \mathbf{A}_{t-1})$ as defined in Algorithm 25, and suppose that $\mathbf{H}_t^{J_t^*}$ represent the valid (or the sufficient) subspace of $\mathbf{H}_t$ for the estimation stage at time t, such that the estimation procedure or the evaluation procedure associates similar risk or evaluation error for the assessed functions, that is,

$$\min_{\substack{f_t:(\mathbf{H}_t,a_t)\mapsto\mathbb{R}\\ f_t \text{ measurable}}} \mathcal{R}_{L,P}(f_t) = \min_{\substack{f_t:(\mathbf{H}_t^{J_t^*},a_t)\mapsto\mathbb{R}\\ f_t \text{ measurable}}} \mathcal{R}_{L,P}(f_t) \tag{4.8}$$

where $\mathcal{R}_{L,P}(f_t)$ denotes the evaluation error or risk of the function f_t. Now since the space of measurable functions are nested, hence $\mathcal{F}_t = \{f_t : (\mathbf{H}_t, a_t) \mapsto \mathbb{R},\, f_t \,\text{measurable}\}$ $\supseteq \mathcal{F}_t^{J_t^*} = \{f_t : (\mathbf{H}_t^{J_t^*}, a_t) \mapsto \mathbb{R},\ f_t \,\text{measurable}\}$ and hence

$$f_{t,P,\mathcal{F}_t} := \arg\min_{f_t\in\mathcal{F}_t} \mathcal{R}_{L,P}(f_t) = \arg\min_{f_t\in\mathcal{F}_t^{J_t^*}} \mathcal{R}_{L,P}(f_t) := f_{t,P,\mathcal{F}_t^{J_t^*}}, \tag{4.9}$$

where P is the oracle probability measure on the input-output space.

Then $\max_{a_t} f_{t,P,\mathcal{F}_t}(\mathbf{H}_t, a_t) = \max_{a_t} f_{t,P,\mathcal{F}_t^{J_t^*}}(\mathbf{H}_t^{J_t^*}, a_t)$. On the other hand in the Q learning setup, under the assumption of 'no unmeasured confounders', if the evaluation error $\mathcal{R}_{L,P}(f_t)$ is the risk associated with predicting the pseudo response $R_t + \max_{a_{t+1}} Q_{t+1}^*$ by the function f_t, then it is easy to see that $f_{t,P,\mathcal{F}_t}$ is the optimal Q-function Q_t^*. And if our belief about the subspace $\mathbf{H}_t^{J_t^*} \subseteq \mathbf{H}_t$ is true, then the assumption of 'no unmeasured confounders' holds true for the space $\mathbf{H}_t^{J_t^*}$ as well and hence $f_{t,P,\mathcal{F}_t^{J_t^*}}$ should also be the optimal Q-function Q_t^*. Hence, it follows that,

$$V_t^*(\mathbf{H}_t) = \max_{a_t} f_{t,P,\mathcal{F}_t}(\mathbf{H}_t, a_t) = \max_{a_t} f_{t,P,\mathcal{F}_t^{J_t^*}}(\mathbf{H}_t^{J_t^*}, a_t) = V_t^*(\mathbf{H}_t^{J_t^*}).$$

This tells us that, the infinite sampled version of the value function remains same if we can meaningfully trim out features that do not contribute towards the outcome, either directly or through interactions with the treatment. Hence, as long as the subset of features we preserve at the end of each run of the elimination mechanism satisfy the 'no unmeasured confounders' assumption, we can also preserve the optimal value function. However, what we observe in practicality (see section 3.10), is that the estimated value function increases monotonically as the size of the history gradually diminishes, as long as we keep all the significant features intact. This behavior is probably due to the high overfitting that is typically present in high-noise models resulting in poor estimation performance. As overfitting decreases, the meaningful signals get magnified, and the estimation performance gets better and hence results in improved estimates of the average value function.

Our goal at the t^{th} stage of Q learning is to characterize the stage t pseudo reward function in terms of variables in $\mathbf{H}_t^{J_t^*}$ meaningfully, so that, for a given patient i with observed history $\mathbf{h}_{t,i}^{J_t^*}$, we can obtain his/her t^{th} optimal treatment by maximizing this pseudo reward function along the treatment rule a_t. Often this maximization depends

on a further subset of features $\mathbf{H}_{t,1}^{J_t^*}$ of $\mathbf{H}_t^{J_t^*}$, which contains features that contribute to the reward function necessarily through interactions with the treatment rule, or in other words, features that are sufficient to fully specify the optimal decision rules. $\mathbf{H}_t^{J_t^*}$ can thus be partitioned into $[\mathbf{H}_{t,1}^{J_t^*}, \mathbf{H}_{t,2}^{J_t^*}]$, such that $d_t^*(\mathbf{H}_{t,1}^{J_t^*}, \mathbf{H}_{t,2}^{J_t^*}) = d_t^*(\mathbf{H}_{t,1}^{J_t^*})$, where $\mathbf{H}_{t,1}^{J_t^*}$ is the minimal subset of $\mathbf{H}_t^{J_t^*}$ satisfying this property. For creating optimal dynamic treatment regimes, we are more interested in $\mathbf{H}_{t,1}^{J_t^*}$, as only features contained in this set help in creating the decision rules.

Now it is not entirely obvious what happens to the value function when a feature belonging to $\mathbf{H}_{t,2}^{J_t^*}$ is removed from the model. We however will come back to this discussion later, but it is not entirely important for our purpose. As we said before, we are more interested in filtering out the set $\mathbf{H}_{t,1}^{J_t^*}$. Now observe that if a feature $\mathcal{X}_0 \in \mathbf{H}_{t,1}^{J_t^*}$ is removed from the model, the decision rule $d_t^*(\mathbf{H}_{t,1}^{J_t^*} \setminus \mathcal{X}_0)$ is necessarily suboptimal, and hence the optimal reward $V_t^*(\mathbf{H}_{t,1}^{J_t^*} \setminus \mathcal{X}_0)$ is suboptimal as well, which would then imply that $V_t^*(\mathbf{H}_{t,1}^{J_t^*}) > V_t^*(\mathbf{H}_{t,1}^{J_t^*} \setminus \mathcal{X}_0)$. Hence we come to an important conjecture needed to develop our third feature selection algorithm, which states that for some $\epsilon_0 > 0$ (specific to the design), the following holds:

$$V_t^*(\mathbf{H}_{t,1}^{J_t^*}) \geq V_t^*(\widetilde{\mathbf{H}}_t) + \epsilon_0,$$

whenever $\widetilde{\mathbf{H}}_t \subset \mathbf{H}_{t,1}^{J_t^*}$. Here is what we believe so far:

- The estimated value function at stage t, $\hat{V}_t$ will increase (or remain the same) if we delete features from the set $\mathbf{H}_t \setminus \mathbf{H}_t^{J_t^*}$.
- $\hat{V}_t$ will decrease if we delete features from the set $\mathbf{H}_{t,1}^{J_t^*}$.

First of all note that the above condition holds for the Q learning algorithm at each individual stage of the trial, and now see that the correct specification of the Q function at each stage (or that of the value function at that stage) depends on the correct

specifications of the optimal rules at all subsequent stages. In Q learning algorithm, the value function at stage t is reached sequentially through iterating the dual framework of estimating the Q function and evaluating the optimal rule (or maximizing the reward function) backwards from stage T till $t+1$, and hence if we misspecify models for the decision rule in any particular stage of the algorithm, not only would the value function for that stage be suboptimal, all estimates of the value functions for the earlier stages would be suboptimal as well. Or in other words, if we evaluate the optimal rule at stage t based on $H \subset \mathbf{H}_{t,1}^{J_t^*}$, and then continue through the subsequent stages of the Q learning algorithm, all our estimates $\{\hat{V}_t^*, \hat{V}_{t-1}^*, \ldots, \hat{V}_1^*\}$ would be suboptimal. This idea allows us to extend our logic to the stage 1 value function, and our belief, that it captures the interactions at all stages sufficiently well, so that the estimated stage 1 value function would increase as if we remove non significant features from the history at any stage of the trial, and that this estimated stage 1 value function would be suboptimal (and hence decrease) if we remove a feature from the history at any particular stage that helps to define the optimal rule at that stage of the trial.

So one important idea in terms of feature selection in Q learning would be to implement the sequential mechanism of the recursive feature elimination algorithm to the estimated value function at stage 1. So at the $k+1^{th}$ stage of the algorithm, given the current history space $\{H_1^{J_{1,k_1}}, \mathbf{H}_2^{J_{2,k_2}}, \ldots, \mathbf{H}_T^{J_{T,k_T}}\}$, such that $|J_{1,k_1}|+|J_{2,k_2}|+\cdots+|J_{T,k_T}| = k$, we construct the new Q-functions on the updated history spaces created by removing one variable at a time from the cumulative history and then estimate the value function $V^*(h_1^{J_{1,k+1_1}})$ at stage 1 for each of these updates. The variable or feature (say feature $X_{(j_{k+1})}$ which originally belonged to history $\mathbf{H}_{t_{(k+1)}}$) for which the empirical expectation of the estimated value function at stage 1 is largest is then eliminated from the system, and the history is revised as $\{H_1^{J_{1,k_1}}, \mathbf{H}_2^{J_{2,k_2}}, \ldots, \mathbf{H}^{J_{t_{(k+1)},k_{t_{(k+1)}}} \cup j_{k+1}}, \ldots, \mathbf{H}_T^{J_{T,k_T}}\}$. The elimination process is continued till the empirical expectation of the estimated value

function at stage 1, $\mathbb{E}_n \hat{V}^k(\cdot)$ attains its maxima.

Now we are ready to give the third algorithm for feature selection in Q learning. To explain the details more clearly, we denote the estimated optimal value function at stage 1, $\hat{V}$ as a function of the entire history to represent the entire input history over which the algorithm computing the optimal value function is conducted. So for history $\{H_1, \mathbf{H}_2, \ldots, \mathbf{H}_T\}$, the estimated optimal value function is given as $\hat{V}(H_1, \mathbf{H}_2, \ldots, \mathbf{H}_T)$.

Algorithm 28 (RFE_Vpred)**.** *Start off with* $J_{1,0}, \ldots, J_{T,0} \equiv [\cdot]$ *empty and the input history set* $\mathcal{H} = \{H_1, \boldsymbol{H}_2, \ldots, \boldsymbol{H}_T\}$. *Let* $Z_1, \ldots, Z_T$ *be the index sets of the variables remaining in the history, such that we can initialize* $Z_t \equiv \{1, 2, \ldots, |H_t|\}$.

Let after k *steps of the algorithm, the updates for the index sets be* $J_{1,k_1}, \ldots, J_{T,k_T}$, *such that* $|J_{1,k_1}| + |J_{2,k_2}| + \cdots + |J_{T,k_T}| = k$. *Let the updates for the input history be* $\mathcal{H}^{J_k} = \{H_1^{J_{1,k_1}}, \boldsymbol{H}_2^{J_{2,k_2}}, \ldots, \boldsymbol{H}_T^{J_{T,k_T}}\}$.

Then at the $k+1^{th}$ *step,*

1. *For* $t \in \{1, 2, \ldots, T\}$ *with* $|Z_t \setminus J_{t,k_t}| > 1$, *find index* $j_{t,k+1}$ *and* $V_{t,k+1}^{max}$ *such that,*

$$j_{t,k+1} = \underset{j_t \in Z_t \setminus J_{t,k_t}}{\arg\max} \mathbb{E}_n \left(\hat{V}^{k+1} \left(h_1^{J_{1,k_1}}, \ldots, h_t^{J_{t,k_t} \cup j_t}, \ldots, h_T^{J_{T,k_T}} \right) \right).$$

$$V_{t,k+1}^{max} = \max_{j_t \in Z_t \setminus J_{t,k_t}} \mathbb{E}_n \left(\hat{V}^{k+1} \left(h_1^{J_{1,k_1}}, \ldots, h_t^{J_{t,k_t} \cup j_t}, \ldots, h_T^{J_{T,k_T}} \right) \right).$$

2. *Now let,*

$$t_{k+1} = \underset{\substack{t \in \{1 \ldots, T\} \\ |Z_t \setminus J_{t,k_t}| > 1}}{\arg\max} V_{t,k+1}^{max},$$

$$V_{k+1}^{max} = \max_{\substack{t \in \{1 \ldots, T\} \\ |Z_t \setminus J_{t,k_t}| > 1}} V_{t,k+1}^{max}.$$

3. Update

$$J_{t_{k+1},k+1_{t_{k+1}}} = J_{t_{k+1},k_{t_{k+1}}} \cup j_{t_{k+1},k+1}.$$
$$J_{t,k+1_t} = J_{t,k_t} \quad \forall t \neq t_{k+1}.$$

And

$$\boldsymbol{H}_{k+1}^{J_{t_{k+1},k+1_{t_{k+1}}}} = \boldsymbol{H}_k^{J_{t_{k+1},k_{t_{k+1}}} \cup j_{t_{k+1},k+1}}.$$
$$\boldsymbol{H}_{k+1}^{J_{t,k+1_t}} = \boldsymbol{H}_k^{J_{t,k_t}} \quad \forall t \neq t_{k+1}.$$

4. If $|Z_t \setminus J_{t,k+1_t}| > 1$ *for any* $t \in \{1, \ldots, T\}$*, go to STEP* 1.

Find k^{stop} *such that*

$$k^{stop} = \arg\max_k V_k^{max}.$$

Now let $k_1^{stop}, \ldots, k_T^{stop}$ *be the subsequent updates for the index sets at the* k^{stop} *step, and output* $J_{1,k_1^{stop}}, \ldots, J_{T,k_T^{stop}}$ *as the sets of indices for features that are to be removed from history* $H_1, \boldsymbol{H}_2, \ldots, \boldsymbol{H}_T$ *respectively.*

4.5 Simulation Results

To determine the performance of the proposed methods for feature selection in Q learning, we conduct simulations under different settings imitating a multistage randomized clinical trial. We roughly follow the simulation settings given in Zhao et al. (2014). We create two scenarios with two-stages and one with three-stages.

4.5.1 Simulation settings

The mechanisms generating the settings are described below:

1. The first setting is a two stage randomized trial with covariate information collected only at baseline.

   - We generate p ($p = 10, 30, 50$) dimensional baseline covariates $X_{1,1}, \ldots, X_{1,p}$ from $N(0, 1)$ and treatments A_1, A_2 are randomly generated from $\{-1, 1\}$ with probability 0.5.
   - Stage 1 outcome R_1 is generated according to $N\left(2X_{1,3}A_1 + 1.5X_{1,4}, 1\right)$.
   - Stage 2 outcome R_2 is generated according to $N((2X_{1,1} + 1.5X_{1,2} + R_1)A_2 + X_{1,5}^2, 1)$.

2. The second setting is also a two stage randomized trial, but with covariate information collected at the start of both stages. Here we incorporate time varying covariates in the stage 2 history. We create two additional binary covariates collected after the first line, the values of which depends on the baseline covariates and the treatment received at stage 1.

   - Like scenario 1, we generate p ($p = 10, 30, 50$) dimensional baseline covariates $X_{1,1}, \ldots, X_{1,p}$ from $N(0, 1)$. Also, similarly treatments A_1, A_2 are randomly generated from $\{-1, 1\}$ with probability 0.5.
   - Stage 1 outcome R_1 is generated in a slightly modified setting from scenario 1, according to $N\left((1 + 1.5X_{1,3})A_1 + X_{1,4}, 1\right)$.
   - Two intermediate variables $X_{2,1} \sim I\{N\left(1.25X_{1,1}A_1, 1\right) > 0\}$ and $X_{2,2} \sim I\{N\left(-1.75X_{1,2}A_1, 1\right) > 0\}$ are generated; Also another $p - 2$ covariates $X_{2,3}, \ldots, X_{2,p}$ are generated from $N(0, 1)$, to make up for the information collected before the beginning of the second stage.
   - Stage 2 outcome R_2 is generated according to $N((0.5 + 1.5R_1 + 1.5A_1 + 2(X_{2,1} - X_{2,2}))A_2 + X_{2,3}, 1)$.

3. In the third scenario, we consider a three-stage SMART design, with the data generating mechanism as follows:

   - We generate p ($p = 10, 30, 50$) dimensional baseline covariates $X_{1,1}, \ldots, X_{1,p}$ from $N(45, 5^2)$, and the treatments A_1, A_2, A_3 are randomly generated from $\{-1, 1\}$ with probability 0.5.
   - For stage 2, one intermediate variable $X_{2,1} \sim N\left(1.5X_{1,1}, 1\right)$ is generated; The rest of the $p-1$ covariates $X_{2,3}, \ldots, X_{2,p}$ are generated from $N(45, 5^2)$.
   - At stage 3, another intermediate variable $X_{3,1} \sim N\left(0.5X_{2,1}, 1\right)$ is generated; The rest $X_{3,3}, \ldots, X_{3,p}$ are again generated from $N(45, 5^2)$.
   - Stage 1 and 2 outcomes $R_1, R_2 = 0$ and R_3 is generated according to $R_3 \sim 20 - |0.6X_{1,1} - 35|\{I(A_1 > 0) - I(X_{1,2} > 45)\}^2 - |0.8X_{2,1} - 60|\{I(A_2 > 0) - I(X_{2,2} > 45)\}^2 - |1.4X_{3,1} - 55|\{I(A_3 > 0) - I(X_{3,2} > 45)\}^2$.

In this scenario, the regret for stage 1 is $|0.6X_{1,1} - 35|\{I(A_1 > 0) - I(X_{1,2} > 45)\}^2$, the regret for stage 2 is given by $|0.8X_{2,1} - 60|\{I(A_2 > 0) - I(X_{2,2} > 45)\}^2$ and the regret for stage 3 is given by $|1.4X_{3,1} - 55|\{I(A_3 > 0) - I(X_{3,2} > 45)\}^2$. We can easily obtain the optimal decision rule by setting the regret to zero at each stage. That is, $d_1^*(h_1) = \text{sign}(x_{1,2} - 45)$, $d_2^*(h_2) = \text{sign}(x_{2,2} - 45)$ and $d_3^*(h_3) = \text{sign}(x_{3,2} - 45)$. In the simulations, we vary sample sizes between 200, 400 and 800, and repeat each scenario 20 times.

The entire methodology was implemented in the MATLAB environment. For the implementation we used the LS-SVMLab library for MATLAB. The LS-SVMLab library can be downloaded from `http://www.esat.kuleuven.be/sista/lssvmlab/`.

4.5.2 Estimation through support vector machines with Gaussian RBF kernel

Q learning is implemented to solve the multistage decision problem, and feature selection is conducted on the Q learning algorithm using the three methods proposed in section 4.3. To estimate the Q functions at each stage, we implement the support vector machines (regression) algorithm with the least squares loss function $L_{LS}(x, y, f(x)) = (y - f(x))^2$. For the optimization of the regressors, we chose the Gaussian RBF kernel $k_\gamma(x_1, x_2) = \exp\{-\frac{1}{\gamma^2}\|x_1 - x_2\|_2^2\}$. We initialize the original SVM function using a 5-fold cross validation on the kernel width γ, chosen from the set of values, 2^{i-3}, $i = \{1, \ldots, 10\}$. and the regularization parameter λ is chosen according to Cherkassky and Ma (2004). The regularization factor λ chosen in this way is much more stable, as opposed to when it is chosen through cross validation, which may often result in excessive overfitting when the input dimension is large compared to the true signals.

In regression, our goal is to find an estimator that has risk as close to the Bayes risk $\mathcal{R}^*_{L,P}$ as possible without being overly complex. The Bayes function or $f_{L,P}$ is the minimizer of risk within $\mathcal{L}_0(\mathcal{X})$, the space of all measurable functions from the input space $\mathcal{X}$ to $\mathbb{R}$. This balance plays an important role in the choice of the kernel for the optimization. The Gaussian RBF kernel generates a very rich RKHS. If $\mathcal{X}$ is compact, the RKHS it produces is dense in the space of all continuous functions $C(\mathcal{X})$ from $\mathcal{X} \mapsto \mathbb{R}$. In fact, it is also dense in the space of all bounded functions on $\mathcal{X}$, $L_\infty(P_\mathcal{X}) = \{f : \mathcal{X} \mapsto \mathbb{R},\ f \text{ bounded}\}$. Now since least squares loss L_{LS} is P-integrable Nemitsky loss, we have the relationship that $\mathcal{R}^*_{L_{LS},P,L_\infty(P_\mathcal{X})} = \mathcal{R}^*_{L_{LS},P}$.[1]

[1] $\mathcal{R}^*_{L_{LS},P,L_\infty(P_\mathcal{X})}$ is the minimized risk attained within the space $L_\infty(P_\mathcal{X})$ for the least squares loss L_{LS}.

In Q learning, the estimation of the stage 1 value function depends on sequentially estimating and maximizing the Q functions from $t = T, T-1, \ldots, 1$. Under misspecification of the models for the Q functions, the estimated stage 1 value function can differ from the true stage 1 value function significantly. For correct implementation of our algorithm, we need to correctly specify the functional relationships in the Q functions. It is thus clear from the discussions in the above paragraph that specifying an RKHS with the Gaussian RBF kernel allows us to closely emulate any complex relationship that the Q functions might have in the feature space.

4.5.3 Stopping rule

Again, the important question we inevitably face in feature elimination is when to stop. Note that for our first method which we also implemented for feature selection for a single stage randomized trial in Dasgupta et al. (2013), we used a change point regression model to obtain the correct set of covariates. The reasoning stemmed from the theoretical standpoint of our derived results for RFE in SVMs that suggested the existence of a gap ϵ_0, and our results further show that asymptotically the difference in the empirical versions of the objective functions exceed this gap whenever we move beyond the correct dimension. Hence if a regression model is fit to the observed objective function values of the algorithm in a scree plot, we will expect a change in the slope of the regression line right after we start eliminating significant covariates because of the aforementioned gap. One plausible way to analyze this gap is to fit a change point regression model of the observed values on the number of cycles of RFE and to infer that the estimated change point is the ad-hoc stopping rule, so as to eliminate all features ranked below that point. For the asymptotic belief that the change in the objective function is negligible to the left of the change point, we can fit a linear trend there. However to the right of the change point, these changes might show non-linear trends, and hence we can fit linear or other polynomial trends to model that. Hence

RFE		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
Stage 1	Prop. no errors (a)	1	1	1	1	1	1	1	1	1
	Prop. 1 error (b)	0	0	0	0	0	0	0	0	0
	Prop. >1 error (c)	0	0	0	0	0	0	0	0	0
Stage 2	Prop. no errors (a)	1	1	0.75	1	1	1	1	1	1
	Prop. 1 error (b)	0	0	0.15	0	0	0	0	0	0
	Prop. >1 error (c)	0	0	0.1	0	0	0	0	0	0
RFE_test		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
Stage 1	Prop. no errors (a)	1	1	1	1	1	1	1	1	1
	Prop. 1 error (b)	0	0	0	0	0	0	0	0	0
	Prop. >1 error (c)	0	0	0	0	0	0	0	0	0
Stage 2	Prop. no errors (a)	1	0.95	0.75	1	1	1	1	1	1
	Prop. 1 error (b)	0	0.05	0.15	0	0	0	0	0	0
	Prop. >1 error (c)	0	0	0.1	0	0	0	0	0	0
RFE_Vpred		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
Stage 1	Prop. no errors (a)	1	0.95	0.8	1	1	1	1	1	1
	Prop. 1 error (b)	0	0.05	0.2	0	0	0	0	0	0
	Prop. >1 error (c)	0	0	0	0	0	0	0	0	0
Stage 2	Prop. no errors (a)	0.8	0.7	0.35	1	1	0.85	1	1	1
	Prop. 1 error (b)	0.2	0.3	0.35	0	0	0.15	0	0	0
	Prop. >1 error (c)	0	0	0.3	0	0	0	0	0	0

Table 4.1: Accuracy of RFE methods in Setting I

this method can be adopted here as well for feature selection at each individual line of treatment.

For the second method using separate data folds for model training and testing, our heuristic belief suggests that overfitted models with high amount of noise tend to produce higher risk estimates on the test data. However, if we can correctly trim out the redundant information and remove the noisy features, that is, if at each stage of the elimination procedure, we can descend down to a subspace of the original feature space that contains the important features, the amount of overfitting diminishes gradually until we reach the only set of features contributing to the relationship. And as argued before, we expect the risk in the test set to start increasing as soon as we start removing any of these significant features. Hence the simple rule for selection of the correct set of covariates in this scenario is to observe the graph of the objective function and choose

RFE		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
Stage 1	Prop. no errors (a)	1	1	0.95	1	1	1	1	1	1
	Prop. 1 error (b)	0	0	0	0	0	0	0	0	0
	Prop. >1 error (c)	0	0	0.05	0	0	0	0	0	0
Stage 2	Prop. no errors (a)	0	0.55	0.3	0	1	0.85	0	1	1
	Prop. 1 error (b)	0.05	0.35	0.45	0	0	0.15	0	0	0
	Prop. >1 error (c)	0.95	0.1	0.25	1	0	0	1	0	0
RFE_test		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
Stage 1	Prop. no errors (a)	1	1	0.95	1	1	1	1	1	1
	Prop. 1 error (b)	0	0	0.05	0	0	0	0	0	0
	Prop. >1 error (c)	0	0	0	0	0	0	0	0	0
Stage 2	Prop. no errors (a)	1	0.6	0.35	1	1	0.85	1	1	0.95
	Prop. 1 error (b)	0	0.35	0.4	0	0	0.15	0	0	0.05
	Prop. >1 error (c)	0	0.05	0.25	0	0	0	0	0	0
RFE_Vpred		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
Stage 1	Prop. no errors (a)	0.8	0.35	0.15	0.9	0.8	0.5	1	1	0.9
	Prop. 1 error (b)	0.2	0.65	0.85	0.1	0.2	0.5	0	0	0.1
	Prop. >1 error (c)	0	0	0	0.04	0	0	0	0	0
Stage 2	Prop. no errors (a)	0.45	0.25	0	0.65	0.4	0.1	0.7	0.55	0.35
	Prop. 1 error (b)	0.35	0.45	0.2	0.35	0.3	0.35	0.3	0.25	0.4
	Prop. >1 error (c)	0.2	0.3	0.8	0	0.3	0.55	0	0.2	0.25

Table 4.2: Accuracy of RFE methods in Setting II

the set of covariates remaining in the model when it attains its minima.

Finally for the third method, again we can implement a very simple stopping rule for our algorithm. As discussed in section 4.4.3 heuristically it is expected that as we move down in the feature space keeping the significant features intact, the regression function (Q function here) minimizing the infinite sampled criterion function (risk, regularized risk or other penalized risks) at each step of the elimination algorithm (and for each stage of the trial) should be the same, and hence the value functions obtained by maximizing these regressors over a given decision rule should be expected to be similar as well. Since we get the stage 1 value function through these sequential estimation and maximization steps, we expect that the stage 1 value function to remain same as we move down in the feature space keeping all significant features intact. However, in the presence of high noise in the model, overfitting can substantially decrease the

empirical estimates of the value function. Hence, as we start removing these redundant features, the amount of overfitting diminishes magnifying the correct relationships and hence increasing the estimated value function. Also as expected the stage 1 value function will start decreasing as soon as we start removing features that directly affect the relationship of the stage 1 value function with the sequence of optimal decision rules. Hence since our final objective is to maximize the stage 1 value function, and that being our criterion of elimination of features as well, the simple rule for selection of the correct set of covariates in this method is to observe the graph of the estimated stage 1 value function and choose the set of covariates remaining in the model when it attains its maxima.

For all three proposed methods in question however, we decide to implement a more conservative selection approach by allowing for a few more features to be selected than the ones obtained using the stopping rules described above. Hence, we allow for a predetermined (and possibly user-defined) percentage to be incorporated with selection mechanism that defines the amount of extra features we want to include in the selection set to safeguard against the chance of losing any important feature on the boundary of the graph. That is, to allow for an $\alpha\%$ error, we select the features given by the stopping rule and additionally allow for another $\lfloor \alpha p \rfloor$ many highest ranked features to be chosen from the remaining ones. Here throughout the different settings, we have allowed for a 5% error rate.

4.5.4 Results

The results of our simulations for Settings 1, 2 and 3 are summarized in Tables 4.1, 4.2 and 4.3 respectively. They display the proportion of times the algorithms (RFE, RFE_test and RFE_Vpred) were able to pick out all the correct features, made only one error, or made multiple errors in their selection sets at each line of treatment for each

artificially created setting. From the results, it is apparent that the first two methods (RFE and RFE_test) work well in all situations, while the third method (RFE_Vpred) struggles most of the times, and we will revisit this later, but for now let us concentrate on the first two methods. A few graphs are given in figures 4.3 – 4.10 at the end of discussion in section 4.7, plotting the objective functions (as opposed to the criterion function or the difference function which is the difference between the objective function in two subsequent steps of the algorithm) for single runs of the algorithm in some of the settings.

RFE & RFE_test: As mentioned before, RFE and RFE_test algorithms are implemented in the estimation phase at each line/stage of treatment of the Q-learning algorithm, and both these methods focus on the correct specification of the Q functions, that is, out of the set of input features, they focus on selecting the entire set of features that correctly specify the Q functions. Setting I being the simplest of the three with covariate information collected only at baseline, both methods work perfectly well (see Table 4.1), except for the $n = 200$, $p = 50$ case, where both of them are prone to some errors, owing to the higher covariate to sample size ratio.

In Setting 2, except for the $p = 10$ case, both these methods perform well and almost at par with each other, with RFE_test marginally dominating the RFE method in some settings. Apparently from the results for this Setting (see Table 4.2), it does seem that RFE fails to work here when the number of covariates is quite small, while RFE_test seem to work perfectly well. This might appear surprising to say the least, but the actual difficulty lie in the manner of choosing the stopping rule. RFE does work in these examples, and does rank the features correctly like the RFE_test method, but the change point model that we fit to select the stopping rule becomes unstable, since the number of features (and hence, the number of observations to fit the change point model) is low, owing to which, it sometimes picks a smaller set of features than the

RFE		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
	Prop. no errors (a)	1	0.95	0.9	1	1	1	1	1	1
Stage 1	Prop. 1 error (b)	0	0.05	0.1	0	0	0	0	0	0
	Prop. >1 error (c)	0	0	0	0	0	0	0	0	0
	Prop. no errors (a)	0.95	0.9	0.4	1	1	1	1	1	1
Stage 2	Prop. 1 error (b)	0.05	0.05	0.25	0	0	0	0	0	0
	Prop. >1 error (c)	0	0.05	0.35	0	0	0	0	0	0
	Prop. no errors (a)	0.7	0.8	0.25	0.75	1	1	0.7	1	1
Stage 3	Prop. 1 error (b)	0.3	0	0.15	0.25	0	0	0.3	0	0
	Prop. >1 error (c)	0	0.2	0.6	0	0	0	0	0	0
RFE_test		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
	Prop. no errors (a)	0.95	0.85	0.75	1	0.95	0.95	1	1	1
Stage 1	Prop. 1 error (b)	0.05	0.1	0.15	0	0.05	0	0	0	0
	Prop. >1 error (c)	0	0.05	0.1	0	0	0.05	0	0	0
	Prop. no errors (a)	0.95	0.7	0.55	1	0.95	0.9	1	1	1
Stage 2	Prop. 1 error (b)	0.05	0.3	0.4	0	0.05	0.1	0	0	0
	Prop. >1 error (c)	0	0	0.25	0	0	0	0	0	0
	Prop. no errors (a)	0.6	0.5	0.25	0.65	0.8	0.75	0.65	1	1
Stage 3	Prop. 1 error (b)	0.3	0.25	0.1	0.35	0.1	0.2	0.35	0	0
	Prop. >1 error (c)	0.1	0.25	0.65	0	0.1	0.1	0	0	0
RFE_Vpred		**n=200**			**n=400**			**n=800**		
		p=10	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**	**p=10**	**p=30**	**p=50**
	Prop. no errors (a)	0.9	0.8	0.6	1	1	0.95	1	1	1
Stage 1	Prop. 1 error (b)	0.1	0.2	0.4	0	0	0.05	0	0	0
	Prop. >1 error (c)	0	0	0	0	0	0	0	0	0
	Prop. no errors (a)	0.9	0.4	0.25	1	0.8	0.15	1	1	0.8
Stage 2	Prop. 1 error (b)	0.1	0.6	0.75	0	0.2	0.85	0	0	0.2
	Prop. >1 error (c)	0	0	0	0	0	0	0	0	0
	Prop. no errors (a)	0.9	0.1	0	1	0.2	0.1	1	0.6	0.2
Stage 3	Prop. 1 error (b)	0.1	0.7	0.35	0	0.7	0.7	0	0.4	0.8
	Prop. >1 error (c)	0	0.2	0.65	0	0.1	0.2	0	0	0

Table 4.3: Accuracy of RFE methods in Setting III

actual size of the correct set (that is, selects a subset of the actual correct set). This is most apparent in the second line, as the number of important features in the model are higher compared to the total number of features in the model (6 out of 23). Figure 4.8(b) actually plots the values of the objective function, and visual inspection does show that the curve changes at the 17^{th} cycle, and hence the rest of the features $(23 - 17 = 6)$ should be chosen as per our belief, but practically speaking, fitting the best change point model (among all linear-quadratic mixtures) doesn't really help to pick out all 6 of them. Probably higher order polynomial mixtures (linear-cubic, linear-quartic or

even higher ones) might help here, but really it becomes a matter of trying different mixtures through trial and error. Hence, we might benefit from a better method to pick out the change in the slope of the objective function for RFE, or in other words, we can use RFE_test. The biggest advantage of this method over the former is its simplified and unambiguous stopping rule. In RFE_test, we select the stopping rule by virtue of observing the objective function, and inspecting the point where it is at its lowest (see section 4.5.3). Figure 4.8(d) does reflect this phenomenon perfectly, and shows why in this scenario, RFE_test performs better than RFE, owing to a better stopping rule. This does make the second method more robust and less sensitive to the number of features in the original model and departures from a perfect separation of the correct set of features from the ones that are superfluous.

In the third setting involving three lines of treatment, both RFE and RFE_test perform relatively well, although performance of RFE dominates that of RFE_test in most scenarios, especially when the sample size is smaller. It is also worthwhile to note that the performance of both methods drop from the first line to the second line, and from second line to the third line, owing to the accumulative effect of history in the Q-learning algorithm. For example in the $p = 50$ case, history at second line has 103 covariates, and at third line has 155 covariates. Hence for smaller sample sizes, both methods struggle in the third line, but the effect diminishes when the sample size increases. This demonstrates that with increased number of lines of treatment, sample size needed to maintain the same level of performance for both methods is also higher. Also another worthy point of mention is both of their relative poor performance in the third lines when the dimension of the covariates is small (when $p = 10$). The reason for this phenomenon is largely unexplained, and steady inspection did show that it is not due to the stopping rule, as was in the previous setting.

RFE_Vpred: RFE_Vpred is implemented in the final evaluation phase of the Q-learning algorithm (maximizing $\hat{Q}_1$, the Q function estimated at first line, to obtain $\hat{V}_1$ or the estimated stage 1 value function), and although the reasoning behind implementation of the stage 1 value function as an elimination criterion is completely ad hoc, but the heuristics for this method are well reasoned and is discussed in detail in Section 4.4.3. The real advantage of this method over the former two, we believe, is in the set of features that this method is able to capture (see a detailed discussion in Appendix B.3.6). While RFE and RFE_test focus on the estimation stage and arguably pick out all features for correct specification of the Q functions, RFE_Vpred focus on the final maximization/evaluation stage and picks out only those features required for correct specification of that part of the Q function which contain only the features involved in decision specific interactions. This aspect sets this method apart from the former two, and can be useful in obtaining features that directly help set up the decision rules at each line. Hence, the performance of this method is evaluated in its ability to pick up the features that interact with the treatment rule, and the results are displayed in Tables 4.1, 4.2 and 4.3. As observed, currently this method does not perform as well as expected, but it does show some promise for future modifications, to develop a more robust and consistent methodology that can achieve the same. It does perform relatively well in Setting I, but the performance gradually deterioriates with increased number of covariates and increased number of lines in the trials (like in Setting II and Setting III) much more drastically than RFE or RFE_test. One important reason for this might be in the current implementation of the algorithm which conducts the elimination procedure over the entire trial and the entire history, and makes it a lot more sensitive than RFE and RFE_test on smaller sample sizes (for a run on the $p = 50$ case in Setting III, it actually performs the elimination algorithm on the accumulated history containing $155 + 103 + 51 = 309$ covariates which becomes difficult in smaller

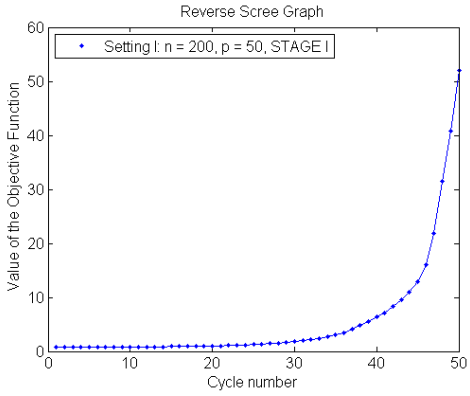
sample sizes like $n = 200,\ 400$). This opens up discussions on a number of interesting modifications that we can try in the future, to make it less sensitive in a way to generate better performance.

Another point to note from the plots displaying the graphs of the objective function for RFE_Vpred (see figure 4.9(g) for example), the estimated average stage 1 value function increases, sometimes sharply, as we continually remove insignificant features from the model, as discussed in section 4.5.3. This makes the stopping rule very intrinsic owing to our final goal of maximizing the stage 1 value function, in choosing the stopping rule at the point when this estimated stage 1 value function reaches its maxima. This observed increase in the value function might be due to high overfitting in the model under high noise to signal ratio, and implies that even under the most general specification of the Q functions, misspecification in the set of features that we include in the model (even if it contains the correct features as a subset) can generate rules that might be suboptimal. This enhances greatly the need for feature selection in Q learning, and establishes the very importance of this project.

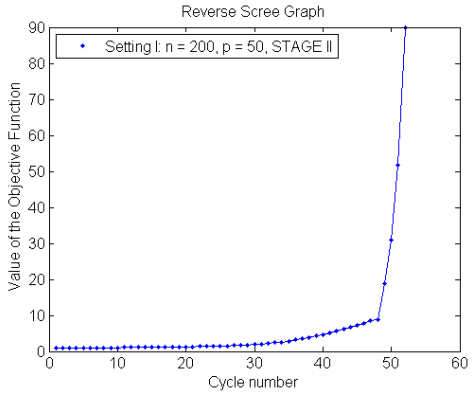
4.6 Summary of Chapter 4

In this work, we focus our attention at a very important aspect of analysis opportunities using these methods, that is, feature selection. With the amount of data available at our disposal these days, feature selection indeed becomes a necessary tool to trim the surplus and redundant information. Here we discussed three different methods for feature selection in Q learning, based on the same vital idea of feature screening through ranking in a sequential backward selection scheme. We discussed the applicability of the methods, reasoned on heuristics stemming from our previous work on feature selection in support vector machines, and gave results showing their performance in various simulated settings.

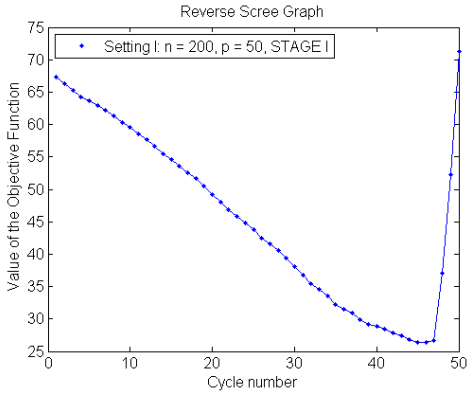
4.7 Plots for single runs of the algorithm in some of the settings



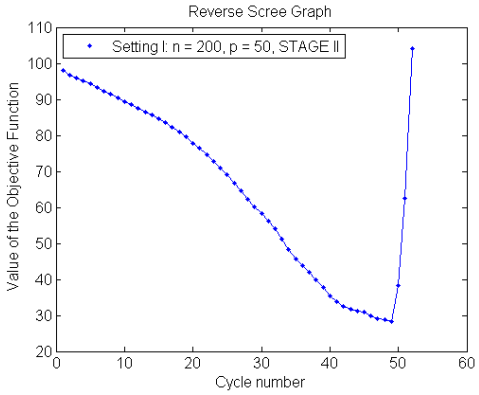
(a) RFE on Stage 1



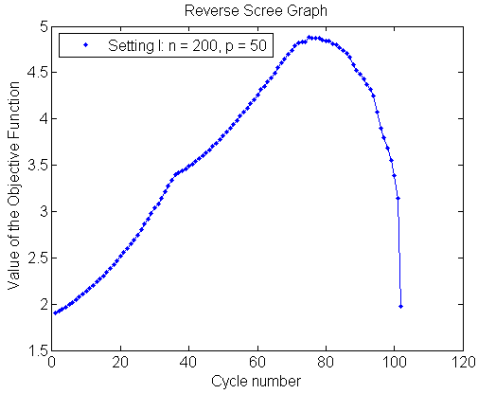
(b) RFE on Stage 2



(c) RFE_test on Stage 1

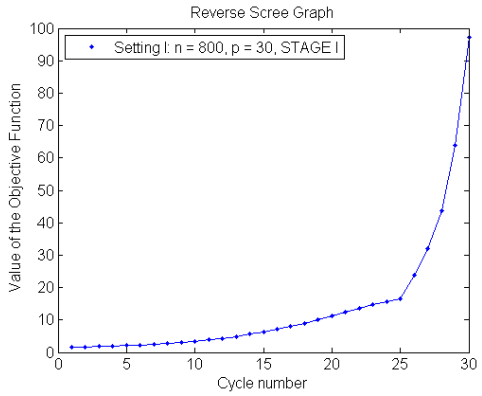


(d) RFE_test on Stage 2

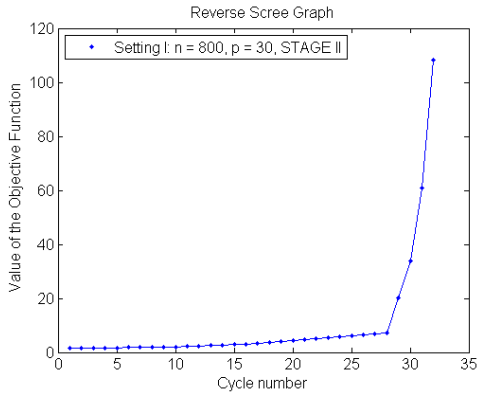


(e) RFE_Vpred

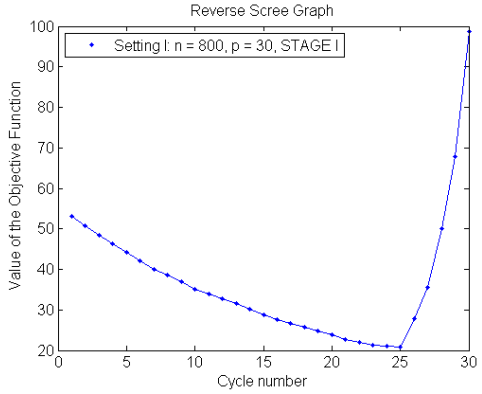
Figure 4.3: Setting I, $n = 200$, $p = 50$



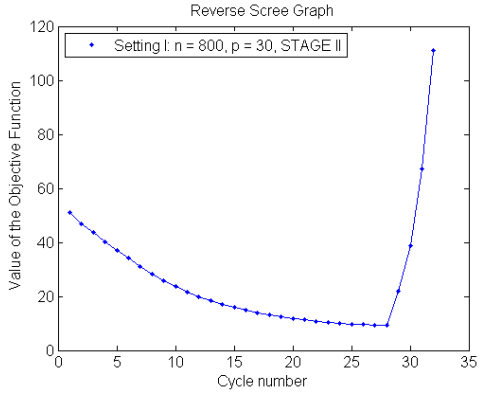
(a) RFE on Stage 1



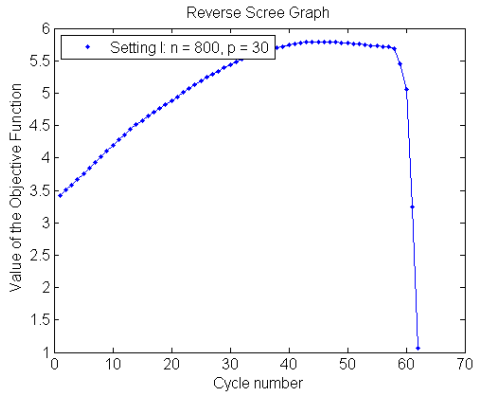
(b) RFE on Stage 2



(c) RFE_test on Stage 1

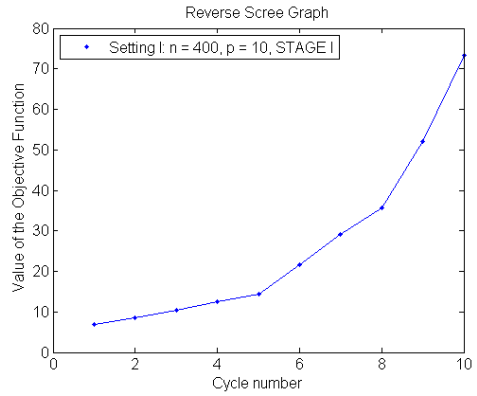


(d) RFE_test on Stage 2

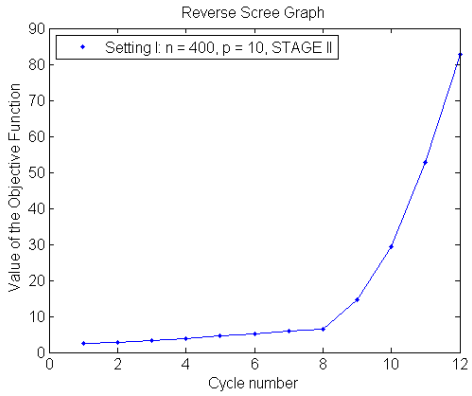


(e) RFE_Vpred

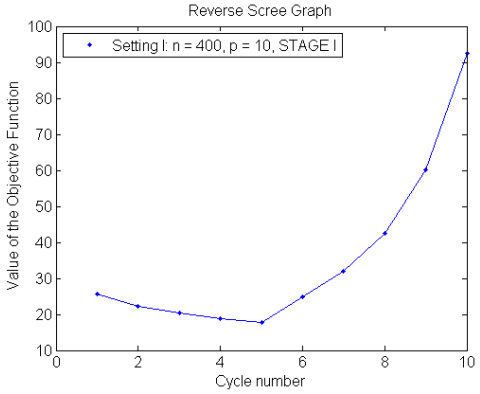
Figure 4.4: Setting I, $n = 800$, $p = 30$



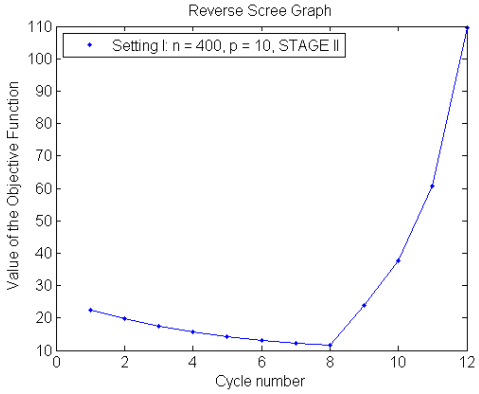
(a) RFE on Stage 1



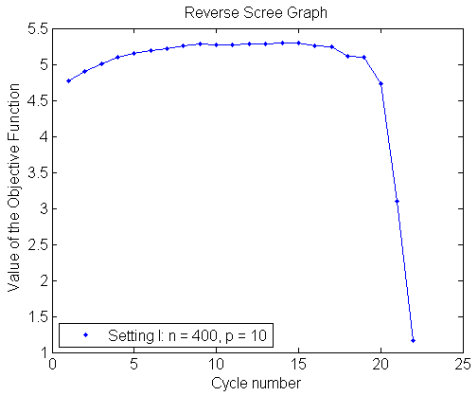
(b) RFE on Stage 2



(c) RFE_test on Stage 1

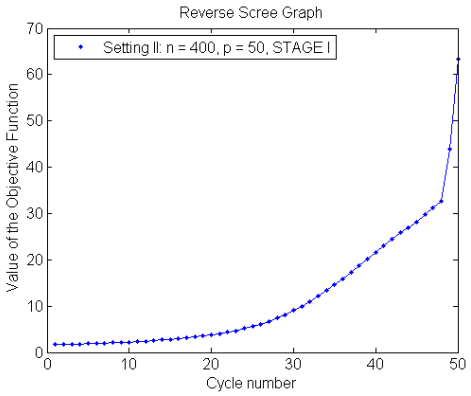


(d) RFE_test on Stage 2

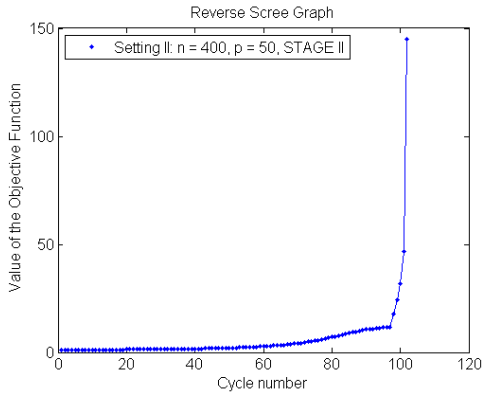


(e) RFE_Vpred

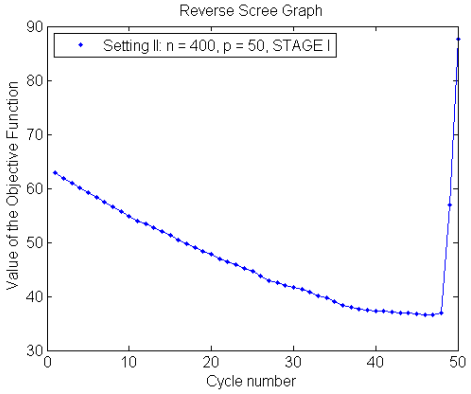
Figure 4.5: Setting I, $n = 400$, $p = 10$



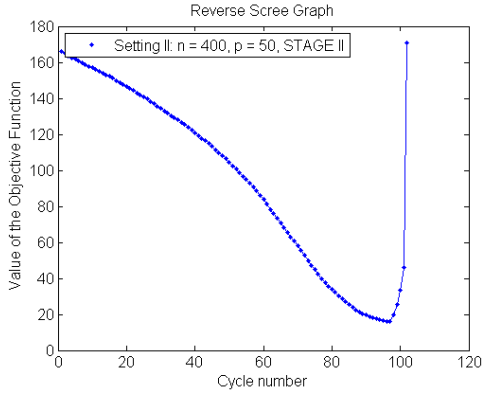
(a) RFE on Stage 1



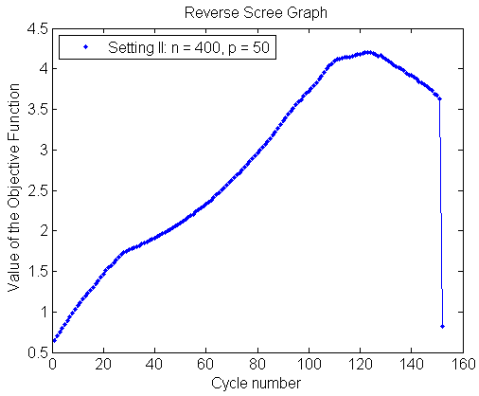
(b) RFE on Stage 2



(c) RFE_test on Stage 1

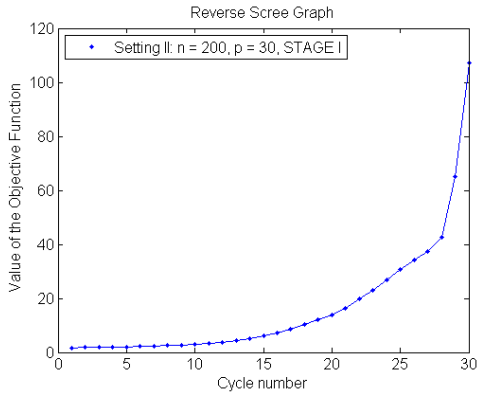


(d) RFE_test on Stage 2

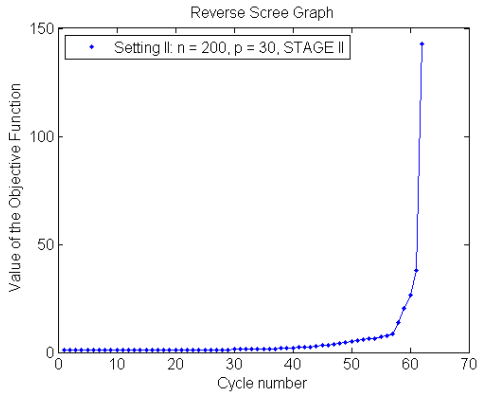


(e) RFE_Vpred

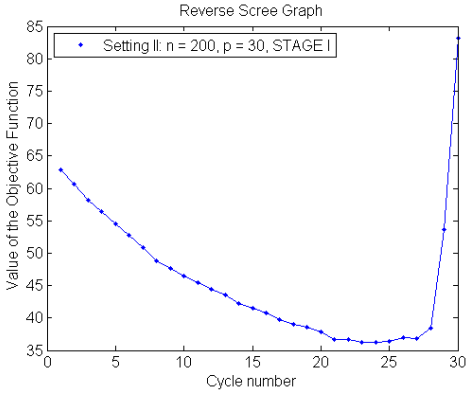
Figure 4.6: Setting II, $n = 400$, $p = 50$



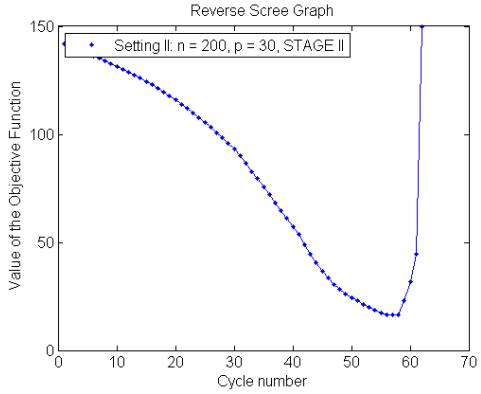
(a) RFE on Stage 1



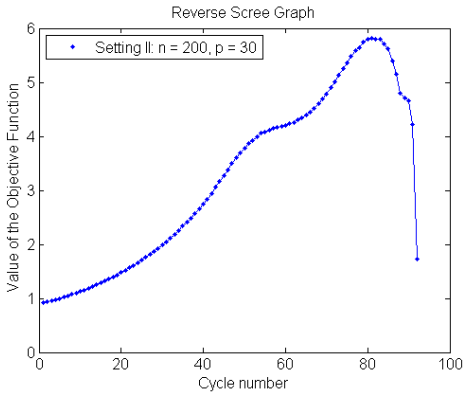
(b) RFE on Stage 2



(c) RFE_test on Stage 1

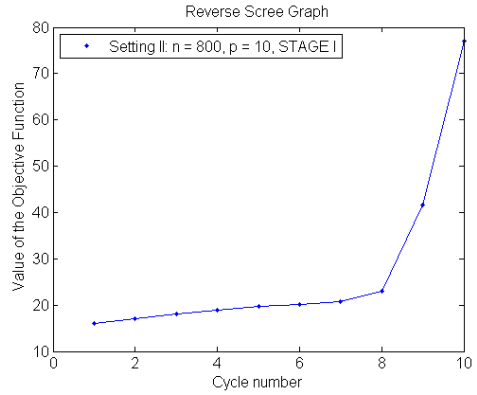


(d) RFE_test on Stage 2

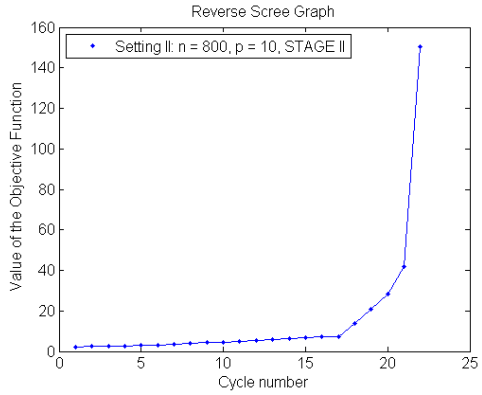


(e) RFE_Vpred

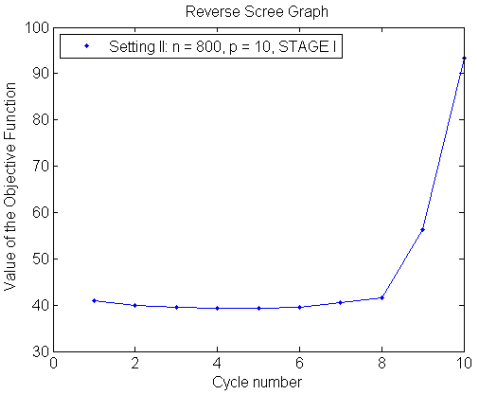
Figure 4.7: Setting II, $n = 200$, $p = 30$



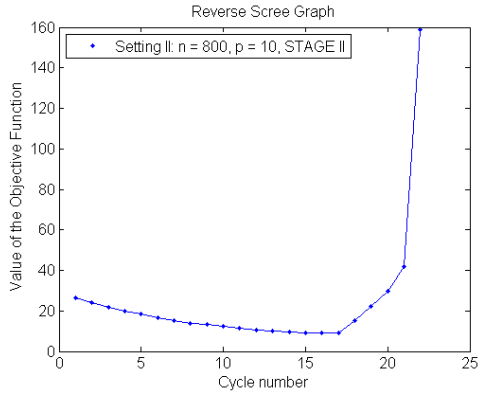
(a) RFE on Stage 1



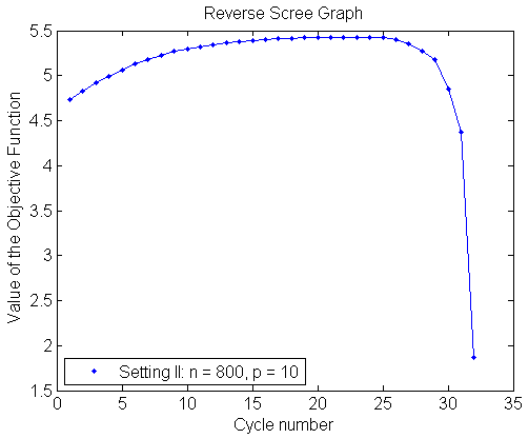
(b) RFE on Stage 2



(c) RFE_test on Stage 1

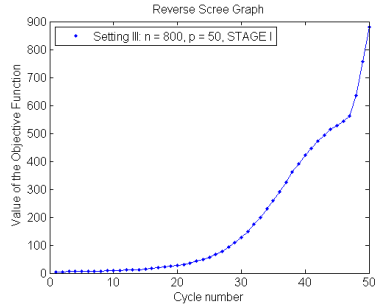


(d) RFE_test on Stage 2

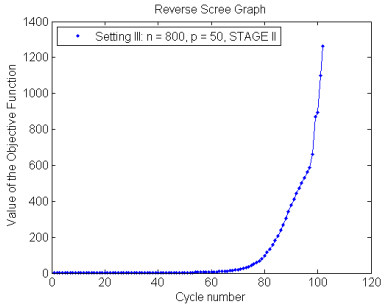


(e) RFE_Vpred

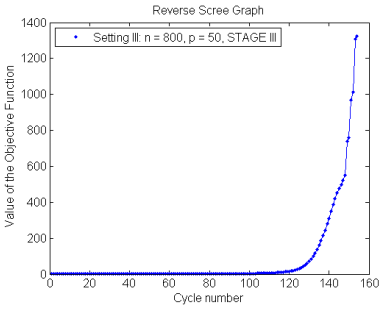
Figure 4.8: Setting II, $n = 800$, $p = 10$



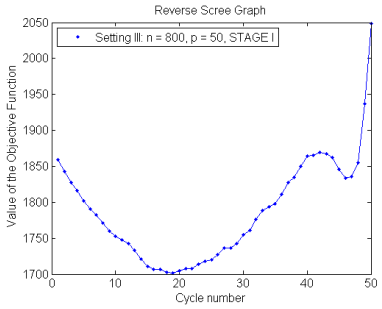
(a) RFE on Stage 1



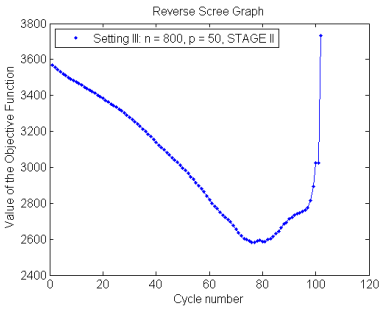
(b) RFE on Stage 2



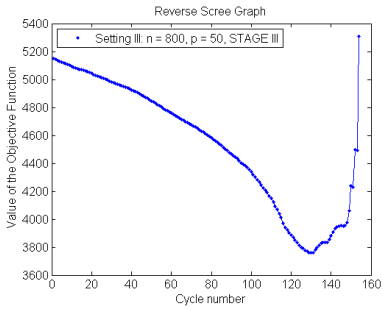
(c) RFE on Stage 3



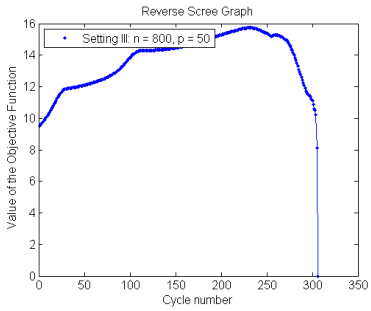
(d) RFE_test on Stage 1



(e) RFE_test on Stage 2

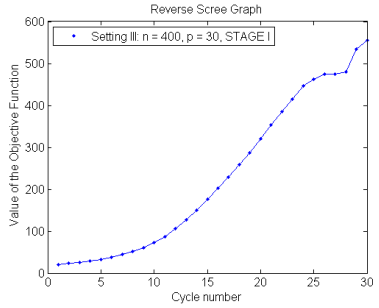


(f) RFE_test on Stage 3

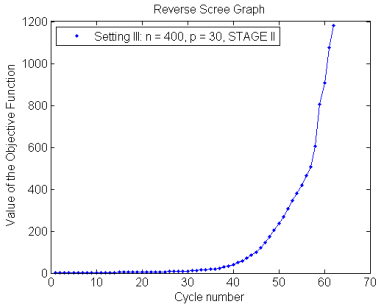


(g) RFE_Vpred

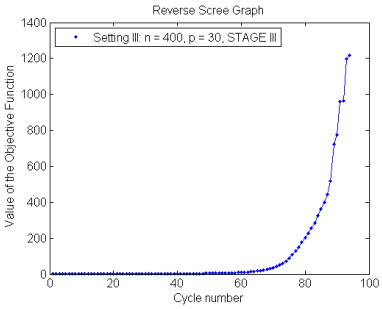
Figure 4.9: Setting III, $n = 800$, $p = 50$



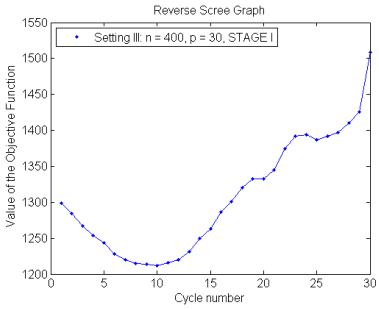
(a) RFE on Stage 1



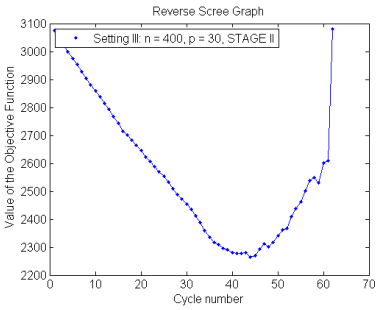
(b) RFE on Stage 2



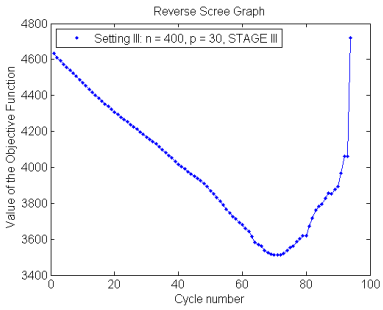
(c) RFE on Stage 3



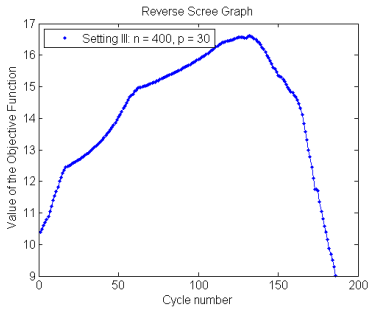
(d) RFE_test on Stage 1



(e) RFE_test on Stage 2

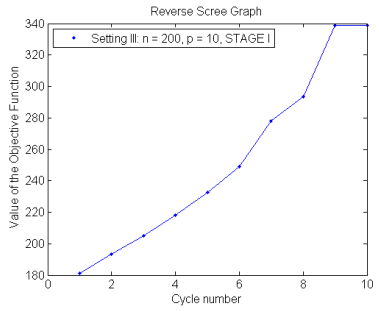


(f) RFE_test on Stage 3

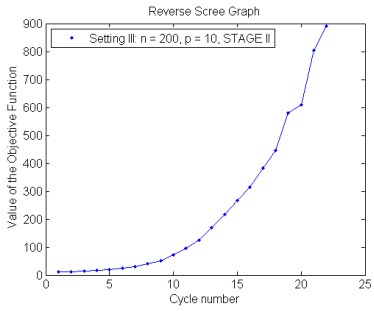


(g) RFE_Vpred

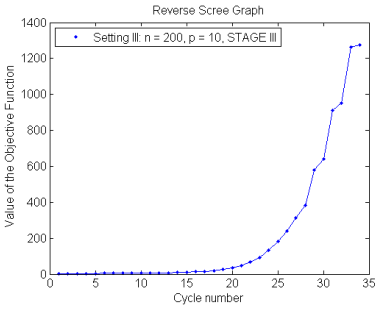
Figure 4.10: Setting III, $n = 400$, $p = 30$



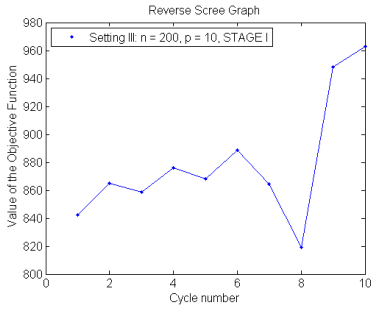
(a) RFE on Stage 1



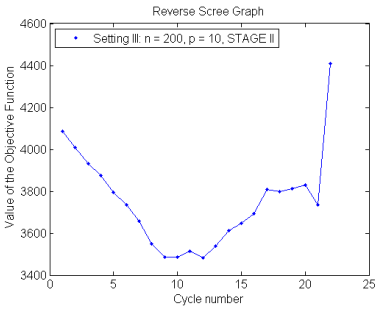
(b) RFE on Stage 2



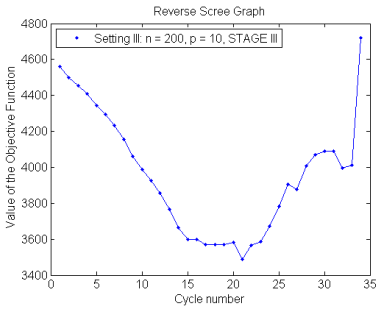
(c) RFE on Stage 3



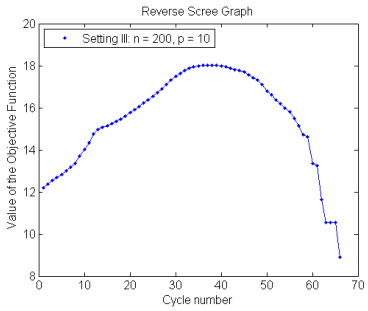
(d) RFE_test on Stage 1



(e) RFE_test on Stage 2



(f) RFE_test on Stage 3



(g) RFE_Vpred

Figure 4.11: Setting III, $n = 200$, $p = 10$

CHAPTER 5: DISCUSSIONS AND FUTURE PROJECTS

In this dissertation, we focused on three different nonparametric and semiparametric methods used in statistical learning. The first project deals with analyzing medical adherence data in Hepatitis C patients using a semiparametric method called temporal process regression. The second and the third projects are related through the common goal of feature selection in two nonparametric methods in vogue today, namely support vector machines and Q learning, respectively.

5.1 Using temporal process regression to study medical adherence

In Chapter 2, the initial analyses showed that adherence to both drugs has a significant effect on the treatment end-point (SVR), with higher adherence significantly increasing the chance of achieving SVR. This confirms the fact that adherence is crucial for effectiveness of the medication regimen for treating chronic hepatitis C. We also found other significant factors that affect SVR. It was seen that women have higher probability of attaining SVR than men. We also saw that race plays an important role in determining chances for a positive drug response and that Caucasians have significantly higher chances of attaining SVR than others. We further saw that the severity of infections (fibrosis score) does affect SVR and patients with higher baseline infection scores have less chances of a full recovery (this reaffirms results found in Conjeevaram et al. (2006)). The combined analysis showed some interesting results as well. The individual effects of the drugs were found significant by the IDS test while the joint effect was found significant by both the IDS and SDS tests. This shows that adherence

to the combined regimen is important to improve chances of achieving SVR, confirming results obtained from the Phase-II drug trials.

Figure 2.8 showed that the effect of interaction between adherence to the drugs can also have a serious impact on SVR. Our results showed that adherence on week 3 has tremendous bearing on the final outcome, which supports the conclusion that adherence in the first few weeks of the regimen is extremely important. This certainly is a new discovery with regards to existing knowledge about adherence in treatment for chronic hepatitis C, and gives a better perception of the temporal relationship of early adherence with the medical end-point SVR. It would thus be interesting to see whether a similar trend is noticed in the proposed triple therapy which is the current point of focus in the medical community for treatment of chronic hepatitis C, and care should be taken to remedy factors that influence early adherence.

Overall, these analyses show a much clearer picture of the relationship between adherence to the drug regimen in the context of achieving a positive end-point after hepatitis C treatment. It reveals trends of this relationship and shows the importance of early adherence in such a context. In addition, it shows that methods used here can be used as a generalize framework for similar analyses in other medical trials and drug regimens. It also illustrates that simply knowing whether adherence is important may not be good enough, and it may be equally important to quantitatively characterize this relationship over the length of the study.

The method we used here does not assume a Markovian structure, and the parameters are interpreted conditionally on covariates at t, and not all $s < t$. Hence the formulation for the conditional mean model will still hold true in absence of a Markov structure, that is, in situations where the response $Y(t)$ depends on covariates at times $s < t$. This might often be true in analyses where adherence is modeled in a temporal framework, conditional on the factor contributing to it. Some of these factors might

have a delayed effect on adherence but that would not hamper the foundation of the functional generalized linear model proposed here. Temporal Process Regression can also be used to magnify these temporal relationships over any subintervals of the actual length of the study and conduct analyses within them. This allows for better understanding of these patterns with respect to different stages of the treatment regime, and allows us to correct for factors that might only affect the response within those intervals in concern. Also as we saw in this analysis, temporal process regression does allow us to model the temporal nature of interactions between factors, like, in our case, interactions between adherence to different drugs in a multi-drug therapy.

However, there are still concerns with regards to usage of these methods in specific situations, and certain necessary assumptions that we inherently make in such a framework. One basic necessity is full availability of data at most of the times of measurements, and higher percentages of missing data may raise a few issues that need to addressed. In certain cases, imputations or other Bayesian or frequentist methods may work well, and in some other cases, like ours, where the response was in fact a measurement done post treatment, assumption of it being constant across the study duration might be a good solution. The hypothesis tests used here (SDS and IDS) work in most situations, but however not any one of them dominates the other in terms of power. SDS might be too conservative in some situations, but it can potentially be more powerful than IDS in others. Hence it is better to use both tests in any given analysis, and to use one of them to re-evaluate results obtained from the other. Also since temporal process regression is a functional version of the generalized linear model, it does suffer from a few parametric assumptions, especially on the link, and the variance function. But as is the case in generalized linear models, misspecifications of these assumptions can be easily remedied by known techniques.

5.2 Consistency results for RFE in SVM

In our second chapter (Chapter 3), we proposed an algorithm for feature elimination in empirical risk minimization and support vector machines. We studied the theoretical properties of the method, discussed the necessary assumptions, and showed that it is universally consistent in finding the correct feature space under these assumptions. We provided case studies of a few of the many different scenarios where this method can be used. Finally, we give a short simulation study to illustrate the method and discuss a practical method for choosing the correct subset of features.

Note that Lemma 20(iii) establishes the existence of a gap in the rate of change of the objective function at the point where our feature elimination method begins removing essential features of the learning problem. This motivated us to use a scree plot of the values of the objective function at each cycle, and indeed our simulation results support our approach by visually exhibiting this gap. Moreover, the graphical interpretation of the scree plot motivated the use of change point regression to select the correct feature space. It would be interesting to conduct a more detailed and formal analysis of this gap in real life settings to facilitate more efficient, automated practical solutions.

As far as our knowledge goes, not much analysis have been done on the properties of variable selection algorithms under such general assumptions on the probability generating mechanisms of the input space, especially in support vector machines. So the results generated in this dissertation can act as a good starting point for similar analyses in other settings. It would also be interesting to analyze RFE for other settings, including censored support vector regression (See Goldberg and Kosorok (2013)) or other machine learning problems, including reinforcement learning (which we study in Chapter 4) or other penalized risk minimization problems.

5.3 Feature selection in Q learning

Reinforcement learning methods are gradually gathering momentum in their applicability in medical research. In Chapter 4, we focus our attention at a very important aspect of analysis opportunities using these methods, that is, feature selection. With the amount of data available at our disposal these days, feature selection indeed becomes a necessary tool to trim the surplus and redundant information. Here we discussed three different methods for feature selection in Q learning, based on the same vital idea of feature screening through ranking in a sequential backward selection scheme. We discussed the applicability of the methods, reasoned on heuristics stemming from our previous work on feature selection in support vector machines, and gave results showing their performance in various simulated settings.

We showed that the first two methods work quite well for feature selection in Q learning. These methods allow feature selection in the estimation phase of the algorithm, and hence they try to retain all of the meaningful signals in the Q functions. As discussed in section 4.5, this means that these methods typically allow to retain all important features necessary for correct specification of the Q functions, but cannot distinguish between features that directly interact with the decision rules in generating the reward, from those that do not. Our simulation results showed that both these methods work quite well, and although using usual RFE over RFE_test might benefit slightly in some situations, we saw that RFE_test is much more robust, and benefits from a natural stopping rule unlike RFE.

We developed the third method RFE_Vpred to utilize the evaluation step for feature selection in Q learning. First of all, this works on the entire algorithm (a backward selection based on the estimated stage 1 value function to be precise), and not sequentially on individual lines of the trial. And second of all, since the elimination is based directly on the stage 1 value function, that is achieved by sequentially estimating and

maximizing along the decision rules, heuristically it should be able to select the features that directly interact with the decision rules. The results do show some promise, however it fails to match the performance of the first two methods. However we do think it is a very important starting point for the problem at hand, that is, to pick out those signals from the history that directly interact with the decision rules to generate rewards, and we do believe ideas developed in creating RFE_Vpred can lead to a more tangible solution in the future.

Also recent methods like A-learning (see Almirall et al. 2005), BOWL/SOWL (see Zhao et al. 2014) have been developed that concentrate only on features that affect the reward functions through interaction with the decision rules. A Learning models only the advantages, $\mu_t = Q_t(H_t, A_t) - V_t(H_t)$ (that is, departure from the t^{th} optimal value function while taking decision rule A_t at time t). Hence it makes fewer assumptions on the underlying data distribution as compared to Q Learning because here only a portion of the true model for Q_t needs to be specified. Modeling only the advantage is analogous to modeling only the decision-specific interaction terms in the regression setting, while leaving the main effects of history H_t unspecified. BOWL/SOWL methods are generalized multi-stage versions of Outcome Weighted Learning (Zhao et al. 2012) that propose to forgo the estimation phase altogether in bid to maximize the optimal rewards directly. Hence these methods also concentrate locally on features interacting with the decision rule in generating rewards. Feature selection is vital in these settings as well, as is evident from our plots in section 4.5, that shows poor performance in estimation of the optimal rewards when the noise to signal ratio is high. Hence, in our future work, it might be interesting to see if RFE_Vpred can be effectively modified to work in these settings, or whether it can motivate to generate more optimal methods for feature extraction in these settings.

APPENDIX A: Technical Details for Chapter 2

We require the following regularity conditions for proving the asymptotic validity of the confidence bands given by (2.2) in Section 2.1.3.

A1 (R_i, X_i, Y_i), $i = 1, \ldots, n$, are i.i.d. and all component processes are cadlag. We require R, X to have total variation over $[l, u]$ bounded by a fixed constant $c < \infty$, and we require Y to have total variation $\tilde{Y}$ over $[l, u]$ with finite second moment.

A2 $t \mapsto \beta(t)$ is cadlag on $[l, u]$.

A3 $h \equiv g^{-1}$ and $\dot{h} = \partial h(u)/(\partial u)$ are Lipschitz continuous and bounded above and below on compact sets.

A4 We require $\inf_{t\in[l,u]} \text{eigmin}\, P[R(t)X(t)X'(t)] > 0$, where *eigmin* denotes the minimum eigenvalue of a matrix.

A5 For all bounded $B \subset \mathbb{R}^p$, the class of random functions $\{V(b, t) : b \in B, t \in [l, u]\}$ is bounded above and below by positive constants and is BUEI (Bounded in uniform entropy integral) and PM (Pointwise measurable). (for detailed discussions on BUEI and PM processes, refer to Sections 9.1.2 and 8.2 of Kosorok (2008)).

First we note the following Lemma.

Lemma 29. *Suppose the class of functions*

$$\{\psi_{\theta,h} - \psi_{\theta_0,h} \ : \ \|\theta - \theta_0\| < \delta, h \in \mathcal{H}\}$$

is P-Donsker for some $\delta > 0$ *and*

$$\sup_{h\in\mathcal{H}} P\left(\psi_{\theta,h} - \psi_{\theta_0,h}\right)^2 \ \to \ 0, \ as\, \theta \to \theta_0.$$

Then if $\hat{\theta}_n \xrightarrow{P} \theta_0$*, we have*

$$\sup_{h \in \mathcal{H}} \left| \tilde{\mathbb{G}}_n \psi_{\hat{\theta}_n, h} - \tilde{\mathbb{G}}_n \psi_{\theta_0, h} \right| = o_P(1),$$

where $\tilde{\mathbb{G}}_n \equiv n^{-1/2} \sum_{i=1}^n \left(\xi_i - \bar{\xi} \right) \delta_{X_i}$ *and* $\xi_1, \ldots, \xi_n \sim$ *i.i.d. mean* 0*, variance* 1 *random variables.*

We omit the proof for Lemma 29 here, as this is a minor modification of Lemma 13.3 of Kosorok (2008). $\mathbb{G}_n \equiv \sqrt{n}\,(\mathbb{P} - P)$ in Lemma 13.3 is replaced by $\tilde{\mathbb{G}}_n$ here, and the proof follows similarly by the multiplier central limit theorem (Theorem 10.4 of Kosorok (2008)).

Theorem 30. *The* $1 - \alpha$*-level simultaneous confidence bands given by* (2.2) *in Section 2.1.3 are asymptotically valid for the true process* $\beta_0(t)$*.*

Proof. Define $A_i^{\gamma}(\beta, t) = R_i(t) D_i'\{\gamma(t)\} V_i\{\gamma(t), t\}[Y_i(t) - h\{\beta'(t) X_i(t)\}]$, where $\gamma,\ \beta \in \{\ell_c^{\infty}([l,u])\}^p$ and $\ell_c^{\infty}(A)$ is the set of real valued bounded functions on A with absolute measure $\leq c$; and $\ell_{\infty}^{\infty}(A) \equiv \ell^{\infty}(A)$. Now let $\mathcal{U} := \{A_1^{\gamma}(\beta, t) :\ \gamma,\ \beta \in \{\ell_c^{\infty}([l,u])\}^p, t \in [l,u]\}$.

Firstly we show that $\mathcal{U}$ is BUEI with square integrable envelope and is PM for each $c < \infty$. For that, first observe that,

$$\{\beta'(t) X(t) :\ \beta \in \{\ell_c^{\infty}([l,u])\}^p, t \in [l,u]\} \text{ and } \{b' X(t) :\ b \in [-c,c]^p, t \in [l,u]\}$$

are equivalent. Next note that Cadlag processes bounded in total variation are both BUEI and PM (by Lemma 22.4 Kosorok (2008)). Now by applying Lemma 9.17 of Kosorok (2008) we have that the class $\mathcal{U}$ is BUEI and PM with square integrable envelope and hence is P-Donsker by Theorem 8.19 of Kosorok (2008).

Next note that from Theorem 22.5 of Kosorok (2008), we have

$$n^{1/2}\{\hat{\beta}(t) - \beta_0(t)\} = n^{-1/2}\sum_{i=1}^{n}\psi_i(t) + o_p^t(1)$$

where $\psi_i(t) = -\{H(t)\}^{-1}A_i\{\beta_0(t), t\}$ is the influence function for the process $\hat{\beta}(t)$ and $H(t) = P\left(R_1(t)D_1'\{\beta_0(t)\}V_1\{\beta_0(t), t\}D_1\{\beta_0(t)\}\right)$. Now see that the class $\{\psi_1(t) : t \in [l, u]\}$ is P-Donsker, since $\mathcal{U}_0 := \{A_i\{\beta_0(t), t\} : t \in [l, u]\}$ is a subclass of the P-Donsker class $\mathcal{U}$ and $H(t)$ is a uniformly bounded measurable function (see Corollary 9.32 of Kosorok (2008)). Then by Theorem 10.4 of Kosorok (2008), we have that $n^{-1/2}\sum_{i=1}^{n}(z_i - \bar{z})\,\psi_i(t) \rightsquigarrow \mathcal{G}$, for z_i's defined in Section 2.1.3, and $\mathcal{G}$, the mean zero Gaussian process and the asymptotic limit of $n^{1/2}\{\hat{\beta}(t) - \beta_0(t)\}$. Now note that,

$$\begin{aligned} n^{-1/2}\sum_{i=1}^{n} z_i\psi_i(t) &= n^{-1/2}\sum_{i=1}^{n}(z_i - \bar{z})\,\psi_i(t) + \bar{z}n^{-1/2}\sum_{i=1}^{n}\psi_i(t) \\ &= n^{-1/2}\sum_{i=1}^{n}(z_i - \bar{z})\,\psi_i(t) + o_P(1) \end{aligned} \tag{5.1}$$

since $n^{-1/2}\sum_{i=1}^{n}\psi_i(t)$ is asymptotically bounded and $\bar{z}$ goes to zero in probability. Hence the whole remainder term goes to zero in probability, and we have,

$$n^{-1/2}\sum_{i=1}^{n} z_i\psi_i(t) \rightsquigarrow \mathcal{G}.$$

Note that Theorem 22.3 of Kosorok (2008) gives us that $\hat{H}(t)$ is the average of i.i.d. processes indexed by $\beta \in \{\ell_c^\infty([l, u])\}^p$ and $t \in [l, u]$, for all $c \geq \sup_{t\in[l,u]}|\beta_0(t)|$ at $\beta := \hat{\beta}$ and all n large enough. These independent and identically distributed processes are themselves P-Glivenko-Cantelli (P-G-C) and sufficiently smooth, which in turn implies that $\sup_{t\in[l,u]}|\hat{H}(t) - H(t)| \to 0$ in probability. From earlier arguments we saw that $\mathcal{U}_0$ is P-Donsker with square-integrable envelope and hence by extension P-G-C as well.

The fact that the class of pair-wise products in a P-G-C class is a P-G-C class by itself (see Corollary 9.32 of Kosorok (2008)) gives us uniform consistency of $\hat{\psi}_1$, $\hat{G}$ and in turn that of $\hat{\Sigma}$. Now note trivially that,

$$n^{-1/2}\sum_{i=1}^{n} z_i\hat{\psi}_i(t) = n^{-1/2}\sum_{i=1}^{n} z_i\psi_i(t) + n^{-1/2}\sum_{i=1}^{n} z_i\left(\hat{\psi}_i(t) - \psi_i(t)\right).$$

Then to show $n^{-1/2}\sum_{i=1}^{n} z_i\hat{\psi}_i(t) \rightsquigarrow \mathcal{G}$, we only need to confirm that

$$n^{-1/2}\sum_{i=1}^{n} z_i\left(\hat{\psi}_i(t) - \psi_i(t)\right) \to 0 \text{ in probability } \forall t \in [l,u]. \tag{5.2}$$

To see this, first note that by recycling arguments given above, we can show that $\{\hat{\psi}_1(t) : t \in [l,u]\}$ lives in a donsker class for n large enough with probability approaching 1, which in turn implies that $\{\hat{\psi}_1(t) - \psi_1(t) : t \in [l,u]\}$ lives in a donsker class for n large enough. Now,

$$\begin{aligned}
&\sup_{s,t\in[l,u]}\left|n^{-1}\sum_{i=1}^{n}\left(\hat{\psi}_i(s) - \psi_i(s)\right)\left(\hat{\psi}_i(t) - \psi_i(t)\right)'\right| \\
&\leq \sup_{s\in[l,u]}\left|n^{-1}\sum_{i=1}^{n}\left(\hat{\psi}_i(s) - \psi_i(s)\right)^{\otimes 2}\right|^{1/2} \sup_{t\in[l,u]}\left|n^{-1}\sum_{i=1}^{n}\left(\hat{\psi}_i(t) - \psi_i(t)\right)^{\otimes 2}\right|^{1/2} \\
&\to 0 \text{ in probability,}
\end{aligned} \tag{5.3}$$

since each term separately converges to 0 in probability. Since $\{\hat{\psi}_1(t) - \psi_1(t) : t \in [l,u]\}$ is a P-Donsker class with bounded square envelopes for large enough n, it is also P-G-C for large enough n, and the preservation properties of P-G-C classes give us that $P\|\hat{\psi}_i(t) - \psi_i(t)\|_\infty^2 \to 0$ in probability for all $t \in [l,u]$.

Hence the conditions of Lemma 29 are satisfied. Now then Lemma 29, along with (5.1) gives us our desired result.

□

APPENDIX B: Technical Details for Chapter 3

Here we give some additional results for Chapter 3.

B.1 Results for RFE in empirical risk minimization

As mentioned before, the results derived for SVMs can easily be extended into the ERM setting.

B.1.1 The Recursive Feature Elimination Algorithm for ERM

For an empirical risk minimization framework with respect to a given functional space $\mathcal{F}$, Algorithm 2 can be modified to match the setting of ERM.

Algorithm 31. *Replace the regularized empirical risk* $\lambda \left\| f_{D,\lambda,H^J} \right\|_{H^J}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right)$ *in Algorithm 2, (defined for a given set of indices* J*) by the empirical risk* $\mathcal{R}_{L,D}\left(f_{D,\mathcal{F}^J}\right)$.

B.1.2 The version of the main result in ERM

Theorem 32. *Let* L *be a convex locally Lipschitz continuous loss function. Let* $\mathcal{F} \subset \mathcal{L}_\infty(\mathcal{X})$ *be non-empty and compact. Let* $M > 0$ *satisfies* $\|f\|_\infty \leq M$, $f \in \mathcal{F}$. *Let* $B > 0$ *be such that it satisfies* $L(x, y, f(x)) \leq B$, $(x, y) \in \mathcal{X} \times \mathcal{Y}$, $f \in \mathcal{F}$. *Assume that for fixed* $n \geq 1$, *there exists constants* $a \geq 1$ *and* $p \in (0, 1)$ *such that* $\mathbb{E}_{D_\mathcal{X} \sim P_\mathcal{X}^n} e_i\left(\mathcal{F}, L_\infty(D_\mathcal{X})\right) \leq a i^{-\frac{1}{2p}}$, $i \geq 1$.

There exists $\{\delta_n\}$ *such that* $\delta_n = \epsilon_0 - O(n^{-\frac{1}{2}})$, *for which the following statements hold:*

1. *The Recursive Feature Elimination Algorithm for empirical risk minimization, defined for* $\{\delta_n\}$ *given above, will find the correct lower dimensional subspace of the input space with probability tending to* 1.

2. *The function chosen by the algorithm achieves the best risk within the original functional space $\mathcal{F}$ asymptotically.*

Note that the above results hold under either of Condition 1 or 2.

B.1.3 Additional results in ERM

Here we provide a few additional results for ERM, similar to the ones we develop for SVM.

Lemma 33. *Let $\mathcal{F} \subset \mathcal{L}_\infty(\mathcal{X})$ be a non-empty functional subspace. Then for any $J \subseteq \{1, 2, \ldots, d\}$,*

1. *If $\mathcal{F}$ is dense in $\mathcal{L}_\infty(\mathcal{X})$, then $\mathcal{F}^J$ is dense in $\mathcal{L}_\infty^J(\mathcal{X})$.*

2. *If $\mathcal{F}$ is compact, then so is $\mathcal{F}^J$.*

3. *$e_i(\mathcal{F}^J, \|.\|_\infty) \leq e_i(\mathcal{F}, \|.\|_\infty)$, $\forall i \geq 1$ where $e_i(\mathcal{F}, \|.\|_\infty)$ is the i^{th} entropy number of the set $\mathcal{F}$ with respect to the $\|.\|_\infty$-norm as defined in Section 3.1.*

The next few results are similar to the ones we develop for support vector machines in Section 3.8. This would set us up to prove Theorem 32.

Proposition 34. *Assume conditions of Theorem 32. For all measurable ERMs and all $\epsilon > 0$, $\tau > 0$, and $n \geq 1$, and for J_1, $J_2 \in \widetilde{\mathcal{J}}$ such that $J_1 \subseteq J_2 \subseteq J_*$, we have with P^n probability $> 1 - e^{-\tau}$,*

$$\left|\mathcal{R}_{L,D}\left(f_{D,\mathcal{F}^{J_2}}\right) - \mathcal{R}_{L,D}\left(f_{D,\mathcal{F}^{J_1}}\right)\right| < 12B\sqrt{\frac{2\tau}{n}} + \frac{20B\tau}{n} + 24K_1\left(\frac{a^{2p}}{n}\right)^{\frac{1}{2}}$$

where $K_1 := \max\left\{B/4, C_1(p)c_L(C)^p B^{1-p}, C_2(p)c_L(C)^{\frac{2p}{1+p}} B^{\frac{1-p}{1+p}}\right\}$.

Consequently we obtain the following two corollaries:

Corollary 35. *Assume conditions of Theorem 32. For any J and all measurable ERMs and all $\epsilon > 0$, $\tau > 0$, and $n \geq 1$, we have with P^n probability $> 1 - e^{-\tau}$,*

$$|\mathcal{R}_{L,D}\left(f_{D,\mathcal{F}^J}\right) - \mathcal{R}^*_{L,P,\mathcal{F}^J}| < 6B\sqrt{\frac{2\tau}{n}} + \frac{10B\tau}{n} + 12K_1\left(\frac{a^{2p}}{\lambda^p n}\right)^{\frac{1}{2}},$$

where K_1 is as before. Additionally if $J \in \widetilde{\mathcal{J}}$, we can replace $\mathcal{R}^*_{L,P,\mathcal{F}^J}$ in the above inequality by $\mathcal{R}^*_{L,P,\mathcal{F}}$.

Corollary 36. Oracle Inequality for ERM: *Assume conditions of Theorem 32. For any J and all $\epsilon > 0$, $\tau > 0$, and $n \geq 1$, we have with P^n probability $> 1 - e^{-\tau}$,*

$$\mathcal{R}_{L,P}\left(f_{D,\mathcal{F}^J}\right) - \mathcal{R}^*_{L,P,\mathcal{F}^J} < 4B\sqrt{\frac{2\tau}{n}} + \frac{20B\tau}{3n} + 8K_1B^{1-p}\left(\frac{a^{2p}}{\lambda^p n}\right)^{\frac{1}{2}},$$

where K_1 is as before.

We now provide Lemma 37 for ERM:

Lemma 37. *Assume conditions of Theorem 32. Then the following statements hold:*

i. For J_1, $J_2 \in \widetilde{\mathcal{J}}$ and $J_1 \subseteq J_2 \subseteq J_*$, $\exists\ (\{\epsilon_n\} > 0) \to 0$ such that we have with P^n probability greater than $1 - 2e^{-\tau}$, $\mathcal{R}_{L,D}\left(f_{D,\mathcal{F}^{J_2}}\right) \leq \mathcal{R}_{L,D}\left(f_{D,\mathcal{F}^{J_1}}\right) + \epsilon_n$.

ii. For $J_1 \in \widetilde{\mathcal{J}}$, $J_2 \notin \widetilde{\mathcal{J}}$ such that $J_1 \subset J_2$, $\exists\ (\{\epsilon_n\} > 0) \to 0$, such that we have with P^n probability greater than $1 - 2e^{-\tau}$, $\mathcal{R}_{L,D}\left(f_{D,\mathcal{F}^{J_2}}\right) > \mathcal{R}_{L,D}\left(f_{D,\mathcal{F}^{J_1}}\right) + \epsilon_0 - \epsilon_n$.

iii. Oracle Property for RFE in ERM: *For a given $J \subseteq \{1, \ldots, d\}$ the infinite-sample risk of the function $f_{D,\mathcal{F}^J}$, $\mathcal{R}_{L,P}\left(f_{D,\mathcal{F}^J}\right)$, converges in measure to $\mathcal{R}^*_{L,P,\mathcal{F}}$ (and hence to $\mathcal{R}^*_{L,P}$ if $\mathcal{F}$ is dense in $\mathcal{L}_\infty(\mathcal{X})$) iff $J \in \widetilde{\mathcal{J}}$.*

B.2 Additional materials on RFE

B.2.1 A further discussion on Projected Spaces

In order to provide a heuristic understanding of the importance of the projection spaces in feature selection, we give an alternative definition of lower dimensional versions of the input space. First, define the map $\sigma^J : \mathbb{R}^d \mapsto \mathbb{R}^{|J|}$ such that for $x = \{x_1, \ldots, x_d\} \in \mathbb{R}^d$, $\sigma^J(x) = \{x_{J_{\min}}, \ldots, x_{J_{max}}\} \in \mathbb{R}^{|J|}$. So $\sigma^J(x)$ is the $|J|$ dimensional vector containing only those elements of x, the coordinates of which are given in the index set J. Hence we can now define the deleted space $\mathcal{X}^{-J}$ as, $\mathcal{X}^{-J} := \{\sigma^{J^c}(x) = \{x_{J^c_{\min}}, \ldots, x_{J^c_{max}}\} : x \in \mathcal{X}\}$.

Now consider the set up of Theorem 7 with $\mathcal{S} \equiv \mathcal{X}^{-J}$ and $\mathcal{X} \equiv \mathcal{X}^J$. We equip $\mathcal{X}^J$ with the restricted kernel k^J, such that $k^J(x, y) = k(x, y)$ for all $x, y \in \mathcal{X}^J$. Now for any $y \in \mathcal{X}^{-J}$, define the map $\varphi \equiv \phi_d^J : \mathcal{X}^{-J} \mapsto \mathcal{X}^J$ as $\phi_d^J(y) = \pi^{J^c}(x)$, where $x \in \mathcal{X}$ satisfies the relation $y = \sigma^{J^c}(x)$. Or in other words the map ϕ_d^J takes an element from the deleted space, fills in the gaps with zeros and returns an element from the projected space. Note then that ϕ_d^J is a bijection, and hence the spaces $\mathcal{X}^J$ and $\mathcal{X}^{-J}$ are isomorphic to each other.

Hence from Theorem 7, we see that $k^J \circ \phi_d^J$ is a kernel defined on $\mathcal{X}^{-J}$ with the corresponding RKHS $H_{k^J \circ \phi_d^J}$. Suppose now that instead of $\mathcal{X}$, our input space is $\mathcal{X}^{-J}$. We want to know whether we can define a kernel, say k^{-J} on $\mathcal{X}^{-J}$, such that it is the natural abridgment of the kernel k on $\mathcal{X}$ (in the sense of being aptly defined on deleted vectors). And in cases when such a natural connect does exist, we want to know if there also exists an inherent connection between k^{-J} and $k^J \circ \phi_d^J$.

The motivation for the definition of k^{-J} stems from previous works on feature elimination in Support Vector Machines. The Recursive Feature Elimination procedure

developed in Guyon et al. (2002) and subsequently revisited and modified in Rakotomamonjy (2003) starts off with a given input space $\mathcal{X}$ and eliminates features using a weight criteria recursively computed by re-training the SVM on the lower dimensional spaces $\mathcal{X}^{-J}$. From their discussion, it is seen that if the Gram matrix of the training vectors $\{x_1, \ldots, x_n\}$ is given by $\{k(x_k, x_j)\}_{k,j=1}^n$, then the Gram matrix of the training vectors $\{x_1^{-i}, \ldots, x_n^{-i}\}$ after deleting a particular variable say $\mathcal{X}_i$ is taken to be $\{k^{-i}(x_k, x_j)\}_{k,j=1}^n$ where $k^{-i}(x_k, x_j) = k(x_k^{-i}, x_j^{-i})$. This clearly takes into account the assumption that the kernel k can be defined on deleted vectors as well, that is, k is well defined for any pair of vectors x and y where $x, y \in \mathbb{R}^{d_0}$ and $d_0 \leq d$. It is intuitively clear that this may not be true for any general kernel k on $\mathbb{R}^d$. Hence we prefer to work with the projected space $\mathcal{X}^J$ instead of the deleted space $\mathcal{X}^{-J}$, as this approach is more general. Through the following lemma however (Lemma 38), we show that in most practical cases (as discussed in Guyon et al. (2002), and Rakotomamonjy (2003)), the kernels we work with satisfy an intrinsic relationship between k^{-J} and $k^J \circ \phi_d^J$ that makes it appropriate to work with either of the setups.

Lemma 38. *For Radial Kernels and Dot Product Kernels,* $k^{-J} = k^J \circ \phi_d^J$.

The proof is simple and therefore omitted.

Also note that for kernels defined on weighted norms, ($k(x, y) = g(\|x - y\|_W)$ where $\|x - y\|_W := (x - y)'W(x - y)$, with W being a positive $d \times d$ diagonal matrix), the above condition is also satisfied.

B.2.2 Entropy Numbers

Let us define the n^{th} entropy number for a metric space. It helps us characterize the complexity of the space and is formally defined as the following:

1. Entropy Numbers: For (T, d) a metric space and for any integer $n \geq 1$, the

$n-$th entropy number of (T, d) is defined as

$$e_n(T,d) := \inf\left\{\epsilon > 0 \ : \ \exists s_1, \ldots, s_{2^{n-1}} \in T \text{ such that } T \subset \bigcup_{i=1}^{2^{n-1}} B_d(s_i, \epsilon)\right\} \tag{5.4}$$

where $B_d(s,\epsilon)$ is the ball of radius ϵ centered at s, with respect to the metric d. If $S : E \mapsto F$ is a bounded linear operator between normed spaces E and F, we write $e_n(S) = e_n(SB_E, \|\cdot\|_F)$, where B_E is the unit ball in E.

B.3 Proofs

B.3.1 Proof of Lemma 3

Proof. The direction $\mathcal{L}_\infty^J(\mathcal{X}^J) \subseteq \mathcal{L}_\infty(\mathcal{X}^J)$ is obvious since co-ordinate projection maps are continuous. To show that $\mathcal{L}_\infty^J(\mathcal{X}^J) \supseteq \mathcal{L}_\infty(\mathcal{X}^J)$ let us take $g \in \mathcal{L}_\infty(\mathcal{X}^J)$. Then $g : \mathcal{X}^J \mapsto \mathbb{R}$ is measurable with $\|g\|_\infty < \infty$. Extend g to $\widetilde{g}$ to include the whole domain $\mathcal{X}$ by defining $\widetilde{g}(x) = g\left(\pi^{J^c}(x)\right)$. Since $\widetilde{g}$ is measurable with $\|\widetilde{g}\|_\infty = \|g\|_\infty$, we have that $\widetilde{g} \in \mathcal{L}_\infty(\mathcal{X})$ and $\widetilde{g} \circ \pi^{J^c} = \widetilde{g}$, so $g = \widetilde{g}\big|_{\mathcal{X}^J} \in \mathcal{L}_\infty^J(\mathcal{X}^J)$. □

B.3.2 Proof of Lemma 33

Proof. (1) For any function $f \in \mathcal{L}_\infty(\mathcal{X})$, by the denseness of $\mathcal{F}$ we can find a sequence of functions $\{g_n\} \in \mathcal{F}$ such that $g_n \to f$ uniformly. Now fix an arbitrary function $f \in \mathcal{L}_\infty^J(\mathcal{X}) \subset \mathcal{L}_\infty(\mathcal{X})$ and consider any sequence of functions $\{g_n\} \in \mathcal{F}$ that converges to f uniformly. Construct the new sequence of functions $\{g_n^J\}$ where for any function $f \in \mathcal{F}$, f^J is defined by $f^J(x) = f(\pi^{J^c}(x))$. Observe trivially that $\{g_n^J\} \in \mathcal{F}^J$.
Now $\{g_n\} \mapsto f$ uniformly $\Rightarrow$ for any $\epsilon > 0$, $\exists\ N$ such that $\forall\ n \geq N$,

$$\begin{aligned} & \sup_{x\in\mathcal{X}} |g_n(x) - f(x)| < \epsilon \quad \forall n \geq N \quad \Rightarrow \sup_{x\in\pi^{J^c}(\mathcal{X})} |g_n(x) - f(x)| < \epsilon \quad \forall n \geq N \\ \Rightarrow \quad & \sup_{x\in\mathcal{X}} |g_n(\pi^{J^c}(x)) - f(\pi^{J^c}(x))| < \epsilon \quad \forall n \geq N \end{aligned}$$

$$\Rightarrow \quad \sup_{x\in\mathcal{X}} |g_n^J(x) - f(x)| < \epsilon \quad \forall n \geq N \qquad (\because \quad f(\pi^{J^c}(x)) = f(x))$$
$$\Rightarrow \quad \{g_n^J\} \mapsto f \quad \text{uniformly.}$$

Hence $\mathcal{F}^J$ is dense in $\mathcal{L}_\infty^J(\mathcal{X})$.

(2) Since $\mathcal{F}$ is compact, for any $\epsilon > 0$, $\exists\ \{f_n\}_{n=1}^{N_\epsilon} \in \mathcal{F}$ such that $\mathcal{F} \subset \bigcup_{n=1}^{N_\epsilon} \mathbb{B}_{\|\cdot\|_\infty}(f_n, \epsilon)$ (where $\mathbb{B}_{\|\cdot\|_\infty}(f_n, \epsilon)$ is a $\|\cdot\|_\infty$ ball of radius ϵ with center f_n). We now fix $f \in \mathcal{F}^J$ and note that $\exists$ an equivalent class of functions $\{g^f\}$ in $\mathcal{F}$ such that for any two functions g_1^f and $g_2^f \in \{g^f\}$ we have that $g_1^f \sim g_2^f$ in the sense that $g_1^f \circ \pi^{J^c} = g_2^f \circ \pi^{J^c} = f$. Fix one such $\widetilde{g}^f \in \{g^f\}$. Since $\widetilde{g}^f \in \mathcal{F}$, $\exists\ f_i \in \{f_n\}_{n=1}^{N_\epsilon}$ such that $d(f_i, \widetilde{g}^f) < \epsilon$, that is,

$$\begin{aligned}
& \sup_{x\in\mathcal{X}} |f_i(x) - \widetilde{g}^f(x)| < \epsilon \quad \Rightarrow \sup_{x\in\pi^{J^c}(\mathcal{X})} |f_i(x) - \widetilde{g}^f(x)| < \epsilon \\
\Rightarrow \quad & \sup_{x\in\mathcal{X}} |f_i(\pi^{J^c}(x)) - \widetilde{g}^f(\pi^{J^c}(x))| < \epsilon \\
\Rightarrow \quad & \sup_{x\in\mathcal{X}} |f_i^J(x) - f(x)| < \epsilon \quad (\because\ \widetilde{g}^f(\pi^{J^c}(x)) = f(x)) \\
\Rightarrow \quad & \{f_n^J\}_{n=1}^{N_\epsilon} \text{ forms a finite } \epsilon\text{-cover for the set } \mathcal{F}^J.
\end{aligned}$$

Hence $\mathcal{F}^J$ is compact.

(3) To see (3), note that if $f_1, \ldots, f_{2^{n-1}}$ is an ϵ-net of $\mathcal{F}$, then for any $f \in \mathcal{F}$, we have $i \in \{1, \ldots, 2^{n-1}\}$ such that $\|f - f_i\|_\infty < \epsilon$. Then,

$$\begin{aligned}
\|f \circ \pi^{J^c} - f_i \circ \pi^{J^c}\|_\infty &= \sup_{x\in\mathcal{X}} \left| f \circ \pi^{J^c}(x) - f_i \circ \pi^{J^c}(x) \right| = \sup_{x\in\mathcal{X}^J} |f(x) - f_i(x)| \\
&\leq \sup_{x\in\mathcal{X}} |f(x) - f_i(x)| = \|f - f_i\|_\infty < \epsilon.
\end{aligned}$$

Hence $f_1 \circ \pi^{J^c}, \ldots, f_{2^{n-1}} \circ \pi^{J^c}$ is an ϵ-net of $\mathcal{F}^J$. □

B.3.3 Proof of Lemma 8

Proof. To see this, let us consider a dot-product kernel k such that $k(x,y) = g(\langle x, y\rangle)$ where $\langle \cdot\,,\, \cdot \rangle$ is the usual Euclidean inner-product. Now consider the pre-RKHSs H_{pre} and H^J_{pre}. We show here that $H^J_{\text{pre}} \subseteq H_{\text{pre}}$ which will imply that $H^J \subseteq H$. To show this, take $f \in H^J_{\text{pre}}$. This implies that f can be written as $f(\cdot) = \sum_{i=1}^{n} \alpha_i k^J(\cdot, x_i)$ for $n \in \mathbb{N}$, $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$, $x_1, \ldots, x_n \in \mathcal{X}$. Hence,

$$\begin{aligned} f(\cdot) &= \sum_{i=1}^{n} \alpha_i k^J(\cdot, x_i) = \sum_{i=1}^{n} \alpha_i k\left(\pi^{J^c}(\cdot), \pi^{J^c}(x_i)\right) = \sum_{i=1}^{n} \alpha_i g\left(\left\langle \pi^{J^c}(\cdot), \pi^{J^c}(x_i)\right\rangle\right) \\ &= \sum_{i=1}^{n} \alpha_i g\left(\left\langle \cdot, \pi^{J^c}(x_i)\right\rangle\right) = \sum_{i=1}^{n} \alpha_i k\left(\cdot, \pi^{J^c}(x_i)\right) \end{aligned}$$

Noting that $\pi^{J^c}(x_1), \ldots, \pi^{J^c}(x_n) \in \mathcal{X}$, we have that $f \in H_{\text{pre}}$. In a similar way, we can show that for any $J_1 \subseteq J_2$, $H^{J_2} \subseteq H^{J_1}$. □

B.3.4 Proof of Lemma 16

Proof. Note that if we define $g_f := L \circ f - E_P(L \circ f)$, then $\mathcal{G} = \{g_f : f \in \mathcal{F}\}$ is a separable Carathéodory set (for a discussion on Carathéodory families of maps, refer to Definition 7.4 in SC08). To see this, first note that $\|g_f\|_\infty \leq \sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} |L \circ f - E_P(L \circ f)| \leq 2B$ for B defined in the statement of the Lemma. Also by assumption, $\|\cdot\|_{\mathcal{F}}$ dominates the pointwise convergence of functions (so $f_n \to f$ in $\|\cdot\|_{\mathcal{F}} \Rightarrow f_n \to f$ pointwise). Then the fact that L is locally-Lipschitz continuous coupled with Lebesgue's Dominated Convergence Theorem (since $\|L \circ f\|_\infty \leq B$) gives us the above assertion.

Now note that $E_P(g_f) = 0$ and $E_P g_f^2 \leq (2B)^2 = 4B^2$ for B as before, so we can apply the Talagrand's Inequality given in Theorem 7.5 of SC08 on G defined as

$G : \mathcal{Z}^n \equiv (\mathcal{X} \times \mathcal{Y})^n \mapsto \mathbb{R}$ such that

$$G(z_1, \ldots, z_n) := \sup_{g_f \in \mathcal{G}} \left| \frac{1}{n} \sum_{j=1}^{n} g_f(z_j) \right| = \sup_{f \in \mathcal{F}} |\mathcal{R}_{L,D}(f) - \mathcal{R}_{L,P}(f)|, \tag{5.5}$$

and hence, for $\gamma = 1$ and for all $\tau > 0$, we have

$$P^n \left(\left\{ z \in \mathcal{Z}^n : G(z) \geq 2E_{P^n}(G) + 2B\sqrt{\frac{2\tau}{n}} + \frac{10B\tau}{3n} \right\} \right) \leq e^{-\tau}. \tag{5.6}$$

So now we need to bound the term $E_{P^n}(G) := E_{P^n} \left\{ \sup_{f \in \mathcal{F}} |\mathcal{R}_{L,D}(f) - \mathcal{R}_{L,P}(f)| \right\}$.

Defining the new Carathéodory set $\mathcal{H}$ as $\mathcal{H} = \{h_f := L \circ f : f \in \mathcal{F}\}$, for a probability distribution P on $\mathcal{Z} \equiv (\mathcal{X} \times \mathcal{Y})$, we can use the idea of symmetrization given in Proposition 7.10 in SC08 to bound $E_{P^n} \left\{ \sup_{f \in \mathcal{F}} |\mathcal{R}_{L,D}(f) - \mathcal{R}_{L,P}(f)| \right\}$. We have for all $n \geq 1$,

$$\begin{aligned} E_{D \sim P^n} \left\{ \sup_{f \in \mathcal{F}} |\mathcal{R}_{L,D}(f) - \mathcal{R}_{L,P}(f)| \right\} &= E_{D \sim P^n} \sup_{h_f \in \mathcal{H}} |E_P h_f - E_D h_f| \\ &\leq 2E_{D \sim P^n} \mathrm{Rad}_D(\mathcal{H}, n), \end{aligned}$$

where $\mathrm{Rad}_D(\mathcal{H}, n)$ is the n-th empirical Rademacher average of the set $\mathcal{H}$ for $D := \{z_1, \ldots, z_n\} \in \mathcal{Z}^n$ with respect to the Rademacher sequence $\{\varepsilon_1, \ldots, \varepsilon_n\}$ and the distribution ν, which is given by $\mathrm{Rad}_D(\mathcal{H}, n) = E_\nu \sup_{h \in \mathcal{H}} \left| \frac{1}{n} \sum_{i=1}^{n} \varepsilon_i h(z_i) \right|$. So we see now that it suffices to bound $E_{D \sim P^n} \mathrm{Rad}_D(\mathcal{H}, n)$.

For that we use theorem 7.16 of SC08, but before that note that the entropy bound means we have for fixed $n \geq 1$, that $\exists$ constants $a \geq 1$ and $p \in (0, 1)$ such that

$$\mathbb{E}_{D_\mathcal{X} \sim P_\mathcal{X}^n} e_i \left(\mathcal{F}, L_\infty(D_\mathcal{X}) \right) \leq a i^{-\frac{1}{2p}}, \qquad i \geq 1. \tag{5.7}$$

First observe that $\mathcal{H} \subset \mathcal{L}_2(P)$. Now since Lipschitz continuity of L gives us that

$|L(x,y,f_1(x)) - L(x,y,f_2(x))|^2 \leq c_L(C)^2 |f_1(x) - f_2(x)|^2$, it is easy to see that $e_i(\mathcal{H}, \|\cdot\|_{L_2(P)}) \leq c_L(C) e_i(\mathcal{F}, \|\cdot\|_{L_2(P_{\mathcal{X}})})$. Hence we have

$$\begin{aligned} E_{D\sim P^n}\left(e_i(\mathcal{H}, \|\cdot\|_{L_2(D)})\right) &\leq c_L(C) E_{D_{\mathcal{X}}\sim P_{\mathcal{X}}^n}\left(e_i(\mathcal{F}, \|\cdot\|_{L_2(D_{\mathcal{X}})})\right) \\ &\leq c_L(C) E_{D_{\mathcal{X}}\sim P_{\mathcal{X}}^n}\left(e_i(\mathcal{F}, \|\cdot\|_{L_\infty(D_{\mathcal{X}})})\right) \\ &\leq c_L(C) a i^{-\frac{1}{2p}}. \end{aligned} \tag{5.8}$$

Now noting that $\|h_f\|_\infty \leq B$ and $E_P h_f^2 \leq B^2$ for B defined as before, the conditions of Theorem 7.16 of SC08 are satisfied with $\widetilde{a} = c_L(C)a$ and hence we have,

$$E_{D\sim P^n} \mathrm{Rad}_D(\mathcal{H}, n) \leq \max\left\{C_1(p)\widetilde{a}^p B^{1-p} n^{-\frac{1}{2}}, C_2(p)\widetilde{a}^{\frac{2p}{1+p}} B^{\frac{1-p}{1+p}} n^{-\frac{1}{1+p}}\right\} \tag{5.9}$$

for constants $C_1(p)$, $C_2(p)$ depending only on p. Hence we finally have, that with probability $\geq 1 - e^{-\tau}$,

$$\begin{aligned} \sup_{f\in\mathcal{F}} |\mathcal{R}_{L,P}(f) - \mathcal{R}_{L,D}(f)| &\leq 2B\sqrt{\frac{2\tau}{n}} + \frac{10B\tau}{3n} \\ &+ \quad 4\max\left\{C_1(p)c_L(C)^p a^p B^{1-p} n^{-\frac{1}{2}}, C_2(p)c_L(C)^{\frac{2p}{1+p}} a^{\frac{2p}{1+p}} B^{\frac{1-p}{1+p}} n^{-\frac{1}{1+p}}\right\}. \end{aligned}$$

That concludes the proof. □

B.3.5 Proof of Proposition 17

Proof. First note that since $B \geq 1$ and $K \geq B^p/4$, we have $24KB^{1-p} \geq 6B > 2$. Now if $a^{2p} > \lambda^p n$, the inequality trivially follows from the fact that

$$\begin{aligned} &\left|\lambda \left\|f_{D,\lambda,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_2}}\right) - \lambda \left\|f_{D,\lambda,H^{J_1}}\right\|_{H^{J_1}}^2 - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_1}}\right)\right| \\ \leq\ &\left|\lambda \left\|f_{D,\lambda,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_2}}\right)\right| + \left|\lambda \left\|f_{D,\lambda,H^{J_1}}\right\|_{H^{J_1}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_1}}\right)\right| \end{aligned}$$

$$\leq \quad 2\mathcal{R}_{L,D}(0) \leq 24KB^{1-p}\left(\frac{a^{2p}}{\lambda^p n}\right)^{\frac{1}{2}},$$

since $\mathcal{R}_{L,D}(0) \leq 1$. Hence we assume from here on that $a^{2p} \leq \lambda^p n$. Now observe that since H is separable, from Lemma 4 we have that the H^Js are also separable. Hence from Lemma 6.23 of SC08 we have that the SVMs produced by these RKHSs are measurable.

Now note that $L(x,y,0) \leq 1 \Rightarrow$ for any distribution Q on $\mathcal{X} \times \mathcal{Y}$, we have that $\mathcal{R}_{L,Q}(0) \leq 1$. Since, $\inf_{f \in H^J} \lambda \|f\|_{H^J}^2 + \mathcal{R}_{L,Q}(f) \leq \mathcal{R}_{L,Q}(0)$, we have that $\left\|f_{Q,\lambda,H^J}\right\|_{H^J} \leq \sqrt{\frac{\mathcal{R}_{L,Q}(0)}{\lambda}}$. Now since by Lemma 4.23 of SC08 $\|f\|_\infty \leq \|k\|_\infty \|f\|_{H^J}$ for all $f \in H^J$, we have that $\left\|f_{Q,\lambda,H^J}\right\|_\infty \leq \left\|f_{Q,\lambda,H^J}\right\|_{H^J} \leq \lambda^{-1/2}$. So, consequently, for every distribution Q on $\mathcal{X} \times \mathcal{Y}$, we have

$$\left|\mathcal{R}_{L,P}\left(f_{Q,\lambda,H^J}\right) - \mathcal{R}_{L,D}\left(f_{Q,\lambda,H^J}\right)\right| \leq \sup_{\|f\|_{H^J} \leq \lambda^{-1/2}} \left|\mathcal{R}_{L,P}(f) - \mathcal{R}_{L,D}(f)\right|. \tag{5.10}$$

Now,

$$\begin{aligned}
&\left|\lambda \left\|f_{D,\lambda,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_2}}\right) - \lambda \left\|f_{D,\lambda,H^{J_1}}\right\|_{H^{J_1}}^2 - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_1}}\right)\right| \\
&\leq \left|\lambda \left\|f_{D,\lambda,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^{J_2}}\right) - \mathcal{R}^*_{L,P,H^{J_2}}\right| \\
&+ \left|\mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_2}}\right) - \mathcal{R}_{L,P}\left(f_{D,\lambda,H^{J_2}}\right)\right| \\
&+ \left|\lambda \left\|f_{D,\lambda,H^{J_1}}\right\|_{H^{J_1}}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^{J_1}}\right) - \mathcal{R}^*_{L,P,H^{J_1}}\right| \\
&+ \left|\mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_1}}\right) - \mathcal{R}_{L,P}\left(f_{D,\lambda,H^{J_1}}\right)\right|,
\end{aligned}$$

since from (A1), $\mathcal{R}^*_{L,P,H^{J_1}} = \mathcal{R}^*_{L,P,H^{J_2}} = \mathcal{R}^*_{L,P,H}$. Noting that
$\lambda \left\|f_{D,\lambda,H^J}\right\|_{H^J}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}^*_{L,P,H^J} \geq 0$, we have from (6.18) of SC08 that

$$\left|\lambda \left\|f_{D,\lambda,H^J}\right\|_{H^J}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}^*_{L,P,H^J}\right|$$

$$\leq A_2^J(\lambda) + \mathcal{R}_{L,P}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right) + \mathcal{R}_{L,D}\left(f_{P,\lambda,H^J}\right) - \mathcal{R}_{L,P}\left(f_{P,\lambda,H^J}\right)$$
$$\leq A_2^J(\lambda) + 2 \sup_{\|f\|_{H^J} \leq \lambda^{-1/2}} \left|\mathcal{R}_{L,P}(f) - \mathcal{R}_{L,D}(f)\right|. \tag{5.11}$$

From (5.10) and (5.11) and the fact that $J_1,\ J_2 \in \widetilde{\mathcal{J}}$ such that $J_1 \subseteq J_2 \subseteq J_*$, we have that

$$\begin{aligned}
&\left|\lambda \left\|f_{D,\lambda,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_2}}\right) - \lambda \left\|f_{D,\lambda,H^{J_1}}\right\|_{H^{J_1}}^2 - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_1}}\right)\right| \\
&\leq A_2^{J_1}(\lambda) + A_2^{J_2}(\lambda) + 3 \sup_{\|f\|_{H^{J_1}} \leq \lambda^{-1/2}} \left|\mathcal{R}_{L,P}(f) - \mathcal{R}_{L,D}(f)\right| \\
&+ 3 \sup_{\|f\|_{H^{J_2}} \leq \lambda^{-1/2}} \left|\mathcal{R}_{L,P}(f) - \mathcal{R}_{L,D}(f)\right|.
\end{aligned}$$

First note that for $f \in \lambda^{-1/2} B_{H^J}$ and $B := c_L(\lambda^{-1/2})\lambda^{-1/2} + 1$, we have $|L(x,y,f(x))| \leq |L(x,y,f(x)) - L(x,y,0)| + L(x,y,0) \leq B$ for all $(x,y) \in \mathcal{X} \times \mathcal{Y}$. Also note that the entropy bound assumption implies that $E_{D_{\mathcal{X}} \sim P_{\mathcal{X}}^n}\left(e_i(\lambda^{-1/2} B_H, \|\cdot\|_{L_\infty(D_{\mathcal{X}})})\right) \leq \lambda^{-1/2} a i^{-\frac{1}{2p}}$.

Now note from Lemma 4 that the conditions of Lemma 16 are satisfied for $\mathcal{F} := \lambda^{-1/2} B_{H^J}$ $\|\cdot\|_{\mathcal{F}} := \|\cdot\|_{H^J}$, $C := \lambda^{-1/2}$ and $B := c_L(\lambda^{-1/2})\lambda^{-1/2} + 1$ for each of the RKHS classes H^J. Also since $a^{2p} \leq \lambda^p n$ and $B \geq 1$, we have $\left(\frac{a^{2p}}{\lambda^p n}\right)^{1/2} \geq \left(\frac{a^{2p}}{\lambda^p n}\right)^{1/(p+1)}$ and $B^{1-p} \geq B^{\frac{1-p}{1+p}}$ for $p \in (0,1)$. Hence we have our assertion. □

B.3.6 Proof of Lemma 20

Proof. (i) Fixing a $\lambda \in [0,1]$, we have that $B := c_L(\lambda^{-1/2})\lambda^{-1/2} + 1 \leq 2\lambda^{-1/2}$. Now since $|X| \leq x \Rightarrow X \leq x$ for any $x \geq 0$, we see from Proposition 17 that

$$\begin{aligned}
&\lambda \left\|f_{D,\lambda,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_2}}\right) - \lambda \left\|f_{D,\lambda,H^{J_1}}\right\|_{H^{J_1}}^2 - \mathcal{R}_{L,D}\left(f_{D,\lambda,H^{J_1}}\right) \\
&< A_2^{J_1}(\lambda) + A_2^{J_2}(\lambda) + 24\lambda^{-1/2}\sqrt{\frac{2\tau}{n}} + 40\lambda^{-1/2}\frac{\tau}{n} + 48K_2\lambda^{-\frac{p-1}{2}}\left(\frac{a^{2p}}{\lambda^p n}\right)^{\frac{1}{2}} \\
&= A_2^{J_1}(\lambda) + A_2^{J_2}(\lambda) + 24\sqrt{2\tau}(\lambda n)^{-\frac{1}{2}} + 40\tau(\lambda^{\frac{1}{2}} n)^{-1} + 48K_2 a^{2p}(\lambda n)^{-\frac{1}{2}}
\end{aligned} \tag{5.12}$$

with probability at least $1-2e^{-\tau}$. Also from Corollary 18, for $J \in \widetilde{\mathcal{J}}$ similarly, we have

$$\begin{aligned}&\left|\lambda \left\|f_{D,\lambda,H^J}\right\|_{H^J}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda,H^J}\right) - \mathcal{R}^*_{L,P,H}\right| \\ &< A_2^J(\lambda) + 12\sqrt{2\tau}(\lambda n)^{-\frac{1}{2}} + 20\tau(\lambda^{\frac{1}{2}}n)^{-1} + 24K_2a^{2p}(\lambda n)^{-\frac{1}{2}} \qquad (5.13)\end{aligned}$$

with probability at least $1-e^{-\tau}$. Now since $\lambda_n \to 0$ and $\lim_{n\to\infty} \lambda_n n = \infty$, Lemma 5.15 along with (5.32) of SC08 gives us that the right hand side of the above inequality converges to 0. So the denseness assumption of the RKHSs additionally gives us the universal consistency of our feature elimination algorithm. To establish the convergence rate of our algorithm we further assume that there exists $c > 0$ and $\beta \in (0, 1]$ such that $A_2^J \leq c\lambda^\beta$ for any J and for all $\lambda \geq 0$. Then it can be seen that asymptotically the best choice for λ_n in (5.12) or (5.13) is a sequence that behaves like $n^{-\frac{1}{(2\beta+1)}}$ and then the inequalities in (5.12) and (5.13) are satisfied with the $l.h.s.$ replaced by ϵ_n and $\epsilon_n/2$ respectively, where ϵ_n is given by $(2c + 24\sqrt{2\tau} + 48K_2a^{2p})n^{-\frac{\beta}{2\beta+1}} + 40\tau n^{-\frac{4\beta+1}{2(2\beta+1)}}$. This proves (i) for $\{\epsilon_n\}$ for a suitable choice of τ.

(ii) Observe from Corollary 18 along with the conditions on λ_n, A_2^J, and steps in the proof of (i) given above that,

$$\left|\lambda_n \left\|f_{D,\lambda_n,H^J}\right\|_{H^J}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^J}\right) - \mathcal{R}^*_{L,P,H^J}\right| < \epsilon_n/2 \qquad (5.14)$$

occurs with P^n probability greater than $1-e^{\tau}$ for any $J \subset \{1, 2, \ldots, d\}$ where ϵ_n is given as before.

Also note that from Assumption (A2) we have that $\mathcal{R}^*_{L,P,H^{J_2}} - \epsilon_0 \geq \mathcal{R}^*_{L,P,H^{J_*}} = \mathcal{R}^*_{L,P,H^{J_1}}$. So for H^{J_2} for $D \in (\mathcal{X} \times (Y))^n$ we have,

$$\begin{aligned}&P^n\left(\left|\lambda_n \left\|f_{D,\lambda_n,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_2}}\right) - \mathcal{R}^*_{L,P,H^{J_2}}\right| < \epsilon_n/2\right) > 1-e^{-\tau} \\ &\Rightarrow P^n\left(\lambda_n \left\|f_{D,\lambda_n,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_2}}\right) + \epsilon_n/2 > \mathcal{R}^*_{L,P,H^{J_2}}\right) > 1-e^{-\tau}, \quad (5.15)\end{aligned}$$

and for H^{J_1} we have

$$\begin{aligned}
&P^n\left(\left|\lambda_n \left\|f_{D,\lambda_n,H^{J_1}}\right\|_{H^{J_1}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_1}}\right) - \mathcal{R}^*_{L,P,H^{J_1}}\right| < \epsilon_n/2\right) > 1 - e^{-\tau} \\
&\Rightarrow P^n\left(\lambda_n \left\|f_{D,\lambda_n,H^{J_1}}\right\|_{H^{J_1}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_1}}\right) < \mathcal{R}^*_{L,P,H^{J_1}} + \epsilon_n/2\right) > 1 - e^{-\tau} \\
&\Rightarrow P^n\left(\lambda_n \left\|f_{D,\lambda_n,H^{J_1}}\right\|_{H^{J_1}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_1}}\right) + \epsilon_0 - \epsilon_n/2 < \mathcal{R}^*_{L,P,H^{J_2}}\right) > 1 - e^{-\tau}.
\end{aligned} \tag{5.16}$$

Then (5.15) and (5.16) from above jointly imply that

$$\lambda_n \left\|f_{D,\lambda_n,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_2}}\right) \tag{5.17}$$

$$> \lambda_n \left\|f_{D,\lambda_n,H^{J_1}}\right\|_{H^{J_1}}^2 + \mathcal{R}_{L,D}\left(f_{D,\lambda_n,H^{J_1}}\right) + \epsilon_0 - \epsilon_n \tag{5.18}$$

with P^n probability greater than $1 - 2e^{-\tau}$.

Also it is easy to see that since $\epsilon_n \to 0$ with $n \to \infty$, the gap $\widetilde{\epsilon}_n = \epsilon_0 - \epsilon_n \longrightarrow \epsilon_0 > 0$.

(iii) From Assumption (A1), Corollary 19, conditions on λ_n, A_2^J, and steps in the proof of (i) given above, the 'if' condition of (iii) follows since for any J and for all $\epsilon > 0$, $\tau > 0$ and $n \geq 1$ we have,

$$P^n\left(D \in (\mathcal{X} \times \mathcal{Y})^n : \left|\lambda_n \left\|f_{D,\lambda_n,H^J}\right\|_{H^J}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda_n,H^J}\right) - \mathcal{R}^*_{L,P,H}\right| < \eta_n\right) > 1 - e^{-\tau}, \tag{5.19}$$

where $\eta_n = (c + 8\sqrt{2\tau} + 16K_2 a^{2p})n^{-\frac{\beta}{2\beta+1}} + 40/3\tau n^{-\frac{4\beta+1}{2(2\beta+1)}}$.

Now for $J_1 \in \widetilde{\mathcal{J}}$ and $J_2 \notin \widetilde{\mathcal{J}}$ we have $\mathcal{R}^*_{L,P,\mathcal{F}^{J_2}} - \epsilon_0 \geq \mathcal{R}^*_{L,P,\mathcal{F}^{J_*}} = \mathcal{R}^*_{L,P,\mathcal{F}^{J_1}}$ and hence the 'only if' condition of (iii) also follows since

$$P^n\left(\lambda_n \left\|f_{D,\lambda_n,H^{J_2}}\right\|_{H^{J_2}}^2 + \mathcal{R}_{L,P}\left(f_{D,\lambda_n,H^{J_2}}\right) - \mathcal{R}^*_{L,P,H^{J_1}} > \epsilon_0 - \eta_n\right) > 1 - e^{-\tau}$$

$$\Rightarrow P^n \left(\lambda_n \left\| f_{D,\lambda_n,H^{J_2}} \right\|^2_{H^{J_2}} + \mathcal{R}_{L,P} \left(f_{D,\lambda_n,H^{J_2}} \right) - \mathcal{R}^*_{L,P,H} > \epsilon_0 - \eta_n \right) > 1 - e^{-\tau}. \quad (5.20)$$

Now since $\eta_n \to 0$ with $n \to \infty$, the gap $\widetilde{\widetilde{\epsilon}}_n = \epsilon_0 - \eta_n \longrightarrow \epsilon_0 > 0$. □

APPENDIX C: Technical Details for Chapter 4

In this section, we discuss more on the mechanisms of the method RFE_Vpred for features that help define the Q functions, but do not interact with the treatment rules in characterizing the rewards. We show here that in the models defined below, if the t^{th} value function is used for feature selection at stage t in the recursive manner typical of the RFE, we may be able to pick out features that only interact with the treatment rule at stage t.

C.1 A further discussion on the mechanisms of RFE_Vpred

As we argued before in section 4.4.3, we believe the following:

- The estimated value function at stage t, $\hat{V}_t$ will increase (or remain the same) if we delete features from the set $\mathbf{H}_t \setminus \mathbf{H}_t^{J_t^*}$.
- $\hat{V}_t$ will decrease if we delete features from the set $\mathbf{H}_{t,1}^{J_t^*}$.

However we did not really discuss what would happen if we delete features from the set $\mathbf{H}_{t,2}^{J_t^*}$, and deferred it for later. We argued that the above observations seem to suggest that if we follow the stopping criterion of our algorithm, we should successfully remove all noisy features (from the set $\mathbf{H}_t \setminus \mathbf{H}_t^{J_t^*}$) and would ultimately reach a subspace of features that would necessarily contain the features in $\mathbf{H}_{t,1}^{J_t^*}$, which for our purpose is good enough.

However our main interest in this method stems from our belief that this method can inherently pick out only features that necessarily characterize the decision rule, i.e. features that affect the reward through interactions with the treatment rule. In this section we will discuss the heuristics that guide our belief which might help in modification of this method to boost its performance. For that however, we need to understand the mechanisms of the algorithm in dealing with features from $\mathbf{H}_{t,2}^{J_t^*}$. Below

we consider two different models for the characterization of the partitions of the history in defining the Q function. For notational ease, we will denote the Q function at t^{th} stage as $f(\cdot)$, and the 'correct' history space at stage t, $\mathbf{H}_t^{J_t^*}$ will be denoted simply as H the partitions $[\mathbf{H}_{t,2}^{J_t^*}, \mathbf{H}_{t,2}^{J_t^*}]$ will be denoted by simply $[H_1, H_2]$ respectively, and the treatment at stage t will be denoted by A (lower case letters will be used to denote realizations of the respective random variables). We will assume H_1 and H_2 to be independent of each other.

1. $\mathcal{F} = \{f : f(h_1, h_2, a) = f_1(h_1, a) + f_2(h_2) + c\}$.

2. $\mathcal{F} = \{f : f(h_1, h_2, a) = f_2(h_2) f_1(h_1, a)\}$.

Note that here $f_1(\cdot)$ actually characterize the decision rule. Suppose the actual solutions in the original space $\mathcal{F}$ are the following:

1. $f_{\mathcal{F}}(h_1, h_2, a) = f_1(h_1, a) + f_2(h_2) + c$.

2. $f_{\mathcal{F}}(h_1, h_2, a) = f_2(h_2) f_1(h_1, a)$.

Now deleting or removing any number of features from history H_2, or for that matter the entire history H_2 amounts to transforming the problem onto the projected space spanned by features only in H_1. In other words, in this case, we look for the constrained solution $f(\cdot)$ inside the space $\mathcal{F}^{J_{H_2}}$. Hence if we remove H_2 from the feature set, we are left with solutions from the projected space $\mathcal{F}^{J_{H_2}}$ of the form

1. $f_{\mathcal{F}^{J_{H_2}}}(h_1, 0, a) = \tilde{f}_1(h_1, a) + \tilde{c}$.

2. $f_{\mathcal{F}^{J_{H_2}}}(h_1, 0, a) = c_1 \tilde{f}_1(h_1, a)$.

Note that for model 1, the intercept term c in (1) denotes the grand mean effect when both H_1 and H_2 are present in the model, while the term $\tilde{c}$ in (1) denotes the grand mean when only H_1 is present. In the least squares formulation of this problem (our

loss function is L_{LS} here), the model grand mean represents the averaged out effect of the functional relationships between all covariates in the system with the response except for the ones present in the model. The solution in the deleted history space (1) would be the one that minimizes infinite sampled risk, that is, we minimize the criterion below:

$$
\begin{aligned}
&\underset{\tilde{c}\in\mathbb{R},\, \tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E\left(Y - \tilde{f}_1(h_1, a) + \tilde{c}\right)^2 \\
=&\underset{\tilde{c}\in\mathbb{R},\, \tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E\left(f_1(h_1, a) + f_2(h_2) + c - \tilde{f}_1(h_1, a) + \tilde{c}\right)^2 \\
=&\underset{\tilde{c}\in\mathbb{R}}{\operatorname{argmin}}\, E\left(f_2(h_2) + c - \tilde{c}\right)^2 + \underset{\tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E\left(f_1(h_1, a) - \tilde{f}_1(h_1, a)\right)^2 \\
&+ \underset{\tilde{c}\in\mathbb{R}}{\operatorname{argmin}}\, E\left(f_2(h_2) + c - \tilde{c}\right) \underset{\tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E\left(f_1(h_1, a) - \tilde{f}_1(h_1, a)\right),
\end{aligned}
$$

where the last term in the last equality follows from our independence assumption. It is easy to see that $\tilde{f}_1 \equiv f_1$ and $\tilde{c} = c + E_{H_2} f_2$ is the solution for the problem in the space $\mathcal{F}^{J_{H_2}}$. It is interesting to note that the function f_1 remains the same for both solutions (before and after removing H_2). Let us now look at the least squares formulation of the problem through model 2, and aim to reach at a solution like before:

$$
\begin{aligned}
&\underset{c_1\in\mathbb{R},\, \tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E\left(Y - c_1\tilde{f}_1(h_1, a)\right)^2 \\
=&\underset{c_1\in\mathbb{R},\, \tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E\left(f_2(h_2)f_1(h_1, a) - c_1\tilde{f}_1(h_1, a)\right)^2 \\
=&\underset{c_1\in\mathbb{R},\, \tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E\left(f_2(h_2)f_1(h_1, a) - c_1 f_1(h_1, a) + c_1 f_1(h_1, a) - c_1\tilde{f}_1(h_1, a)\right)^2 \\
=&\underset{c_1\in\mathbb{R},\, \tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E_{H_1} f_1(h_1, a) E_{H_2}\left(f_2(h_2) - c_1\right)^2 \\
&+ \underset{c_1\in\mathbb{R},\, \tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad E_{H_1}\left[c_1\left(f_1(h_1, a) - \tilde{f}_1(h_1, a)\right) f_1(h_1, a) E_{H_2}\left(f_2(h_2) - c_1\right)\right] \\
&+ \underset{c_1\in\mathbb{R},\, \tilde{f}_1 \text{ measurable}}{\operatorname{argmin}} \quad c_1 E_{H_1}\left(f_1(h_1, a) - \tilde{f}_1(h_1, a)\right)^2
\end{aligned}
$$

where again the last term in the last equality follows from our independence assumption. Now it is easy to see $\tilde{f}_1 \equiv f_1$ and $c_1 = E_{H_2} f_2$ is the solution for the problem in the space $\mathcal{F}^{J_{H_2}}$. Let us now look at the behavior of the expectation of the t^{th} stage value function V_t when we remove H_2 from the history.

For model 1, $E_{H_t} V_t$ is given by the following:

- Keep H_2 intact: $E_H \max_a f_{\mathcal{F}}(h_1, h_2, a) = E_{H_1} \max_a f_1(h_1, a) + E_{H_2} f_2(h_2) + c$.
- Delete H_2: $E_H \max_a f_{\mathcal{F}^{J_{H_2}}}(h_1, 0, a) = E_{H_1} \max_a f_1(h_1, a) + \tilde{c}$.

And for model 2, $E_{H_t} V_t$ is given by the following:

- Keep H_2 intact: $E_H \max_a f_{\mathcal{F}}(h_1, h_2, a) = E_{H_2} f_2(h_2) E_{H_1} \max_a f_1(h_1, a)$.
- Delete H_2: $E_H \max_a f_{\mathcal{F}^{J_{H_2}}}(h_1, 0, a) = c_1 E_{H_1} \max_a f_1(h_1, a)$.

In lieu of our discussions above, it is now easy to see that in both models 1 and 2, the expected value function for stage t remains the same when we delete features from H_2, which in lieu of our discussion in section 4.4.3 means only the features in H_1 will remain in the model if we use the stage t value function to eliminate features from history at stage t.

This shows in an ad hoc sense, why using the t^{th} value function for feature selection at stage t might be useful in picking out features that only interact with the treatment rule at stage t. Our method currently uses the stage 1 value function to delete features from the entire history, which is much more complicated than what this formulation suggests. A possible modification of this method by using the value functions at each stage to select features at individual stages of the trial might be helpful in the future in increasing its performance.

REFERENCES

Aksu, Y. (2012), "A Fast SVM-based Feature Selection Method, Combining MFE (Margin-Maximizing Feature Elimination) and Upper Bound on Misclassification Risk," *Unpublished manuscript*.

Aksu, Y., Miller, D. J., Kesidis, G., and Yang, Q. X. (2010), "Margin-Maximizing Feature Elimination Methods for Linear and Nonlinear Kernel-Based Discriminant Functions," *IEEE Transactions on Neural Networks*, 21, 701–717.

Almirall, D., Gunter, L. L., and Murphy, S. A. (2005), "Efficient a-learning for dynamic treatment regimes: a handout," .

Bellman, R. (1956), "Dynamic programming and the smoothing problem," *Management Science*, 3, 111–113.

Blatt, D., Murphy, S. A., and Zhu, J. (2004), "A-learning for approximate planning," *Ann Arbor*, 1001, 48109–2122.

Bradley, P. S. and Mangasarian, O. L. (1998), "Feature Selection via Concave Minimization and Support Vector Machines," in *Machine Learning Proceedings of the Fifteenth International Conference(ICML 98)*, Morgan Kaufmann, pp. 82–90.

Chan, A. B., Vasconcelos, N., and Lanckriet, G. R. G. (2007), "Direct convex relaxations of sparse SVM," in *Proceedings of the 24th international conference on Machine learning*, ACM, ICML '07, pp. 145–153.

Chapelle, O., Haffner, P., and Vapnik, V. N. (1999), "Support vector machines for histogram-based image classification," *Neural Networks, IEEE Transactions on*, 10, 1055–1064.

Chapelle, O., Vapnik, V., Bousquet, O., and Mukherjee, S. (2002), "Choosing Multiple Parameters for Support Vector Machines," *Machine Learning*, 46, 131–159.

Cherkassky, V. and Ma, Y. (2004), "Practical selection of SVM parameters and noise estimation for SVM regression," *Neural networks*, 17, 113–126.

Conjeevaram, H. S., Fried, M. W., Jeffers, L. J., Terrault, N. A., Wiley-Lucas, T. E., Afdhal, N., Brown, R. S., Belle, S. H., Hoofnagle, J. H., Kleiner, D. E., Howell, C. D., and Virahep-C-Study-Group (2006), "Peginterferon and ribavirin treatment in African American and Caucasian American patients with hepatitis C genotype 1." *Gastroenterology*, 131, 470.

Dasgupta, S., Goldberg, Y., and Kosorok, M. R. (2013), "Feature selection in empirical risk minimization and support vector machines," Under revision from *The Annals of Statistics*.

Evon, D. M., Esserman, D. A., Bonner, J. E., Rao, T., Fried, M. W., and Golin, C. E. (2013), "Adherence to PEG/ribavirin treatment for chronic hepatitis C: prevalence, patterns, and predictors of missed doses and nonpersistence," *Journal of Viral Hepatitis*.

Fine, J. P., Yan, J., and Kosorok, M. R. (2004), "Temporal process regression," *Biometrika*, 91, 683–703.

Flume, P. A., O'Sullivan, B. P., Robinson, K. A., Goss, C. H., Mogayzel Jr, P. J., Willey-Courand, D. B., Bujan, J., Finder, J., Lester, M., Quittell, L., Rosenblatt11, R., Vender, R. L., Hazle, L., Sabadosa, K., and Marshall, B. (2007), "Cystic fibrosis pulmonary guidelines: chronic medications for maintenance of lung health," *American journal of respiratory and critical care medicine*, 176, 957–969.

Goldberg, Y. and Kosorok, M. R. (2013), "Support Vector Regression for Right Censored Data," *Submitted to Annals of Stat*.

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002), "Gene Selection for Cancer Classification using Support Vector Machines," *Machine Learning*, 46, 389–422.

Hastie, T. and Tibshirani, R. (1993), "Varying-coefficient models," *Journal of the Royal Statistical Society. Series B (Methodological)*, 757–796.

Hastie, T., Tibshirani, R., and Friedman, J. (2001), "The elements of Statistical Learning: data mining, inference, and prediction," *Springer*.

Iqbal, M. J., Faye, I., Samir, B. B., and Said, A. M. (2014), "Efficient Feature Selection and Classification of Protein Sequence Data in Bioinformatics," *The Scientific World Journal*, 2014.

Jolliffe, I. T. (2002), *Principal Component Analysis*, Springer.

Kosorok, M. R. (2008), *Empirical Processes and Semiparametric Inference*, Springerverlag New York.

Krzakowski, M., Ramlau, R., Jassem, J., Szczesna, A., Zatloukal, P., Von Pawel, J., Sun, X., Bennouna, J., Santoro, A., Biesma, B., Delgado, F. M., Salhi, Y., Vaissiere, N., Hansen, O., Tan, E., Quoix, E., Garrido, P., and Douillard, J. Y. (2010), "Phase III trial comparing vinflunine with docetaxel in second-line advanced non–small-cell lung cancer previously treated with platinum-containing chemotherapy," *Journal of Clinical Oncology*, 28, 2167–2173.

Lavori, P. W. and Dawson, R. (2000), "A design for testing clinical strategies: biased adaptive within-subject randomization," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 163, 29–38.

— (2004), "Dynamic treatment regimes: practical design considerations," *Clinical trials*, 1, 9–20.

Leslie, C. S., Eskin, E., Cohen, A., Weston, J., and Noble, W. S. (2004), "Mismatch string kernels for discriminative protein classification," *Bioinformatics*, 20, 467–476.

Liang, K. Y. and Zeger, S. L. (1986), "Longitudinal data analysis using generalized linear models," *Biometrika*, 73, 13–22.

Lin, D. Y., Fleming, T. R., and Wei, L. J. (1994), "Confidence bands for survival curves under the proportional hazards model," *Biometrika*, 81, 73–81.

Liu, H., Golin, C. E., Miller, L. G., Hays, R. D., Beck, C. K., Sanandaji, S., Christian, J., Maldonado, T., Duran, D., Kaplan, A. H., , and Wenger, N. S. (2001), "A comparison study of multiple measures of adherence to HIV protease inhibitors," *Annals of Internal Medicine*, 134, 968–977.

Liu, Y., Helen Zhang, H., Park, C., and Ahn, J. (2007), "Support vector machines with adaptive L_q penalty," *Computational Statistics and Data Analysis*, 51, 6380–6394.

Liu, Y. and Wu, Y. (2007), "Variable Selection via a Combination of the l_0 and l_1 Penalties," *Journal of Computational & Graphical Statistics*, 16, 782–798.

Micchelli, C. A., Xu, Y., Zhang, H., and Lugosi, G. (2006), "Universal kernels," *J. Machine Learning Research*, 7, 2651–2667.

Mladenic, D., Brank, J., Grobelnik, M., and Milic-Frayling, N. (2004), "Feature selection using linear classifier weights: interaction with classification models," in *SIGIR*, pp. 234–241.

Moodie, E. E. M., Richardson, T. S., and Stephens, D. A. (2007), "Demystifying optimal dynamic treatment regimes," *Biometrics*, 63, 447–455.

Murphy, S. A. (2003), "Optimal dynamic treatment regimes," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65, 331–355.

— (2005a), "An experimental design for the development of adaptive treatment strategies," *Statistics in medicine*, 24, 1455–1481.

— (2005b), "A generalization error for Q-learning," *Journal of machine learning research: JMLR*, 6, 1073.

Murphy, S. A., Oslin, D. W., Rush, A. J., and Zhu, J. (2006), "Methodological challenges in constructing effective treatment sequences for chronic psychiatric disorders," *Neuropsychopharmacology*, 32, 257–262.

Murphy, S. A., Van Der Laan, M. J., and Robins, J. M. (2001), "Marginal mean models for dynamic regimes," *Journal of the American Statistical Association*, 96, 1410–1423.

Osterberg, L. and Blaschke, T. (2005), "Adherence to medication," *New England Journal of Medicine*, 353, 487–497.

Paulsen, V. I. (2009), "An Introduction to the theory of Reproducing Kernel Hilbert Spaces," *Unpublished notes*.

Peng, H., Long, F., and Ding, C. (2005), "Feature Selection based on mutual information: Criteria of Max dependency, max-relevance, and min-dependency," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27, 1226–1238.

Radloff, L. S. (1977), "The CES-D scale A self-report depression scale for research in the general population," *Applied psychological measurement*, 1, 385–401.

Rakotomamonjy, A. (2003), "Variable Selection Using SVM-based Criteria," *Journal of Machine Learning Research*, 3, 1357–1370.

Ramsay, J. O. and Dalzell, C. J. (1991), "Some tools for functional data analysis," *Journal of the Royal Statistical Society. Series B (Methodological)*, 539–572.

Robins, J. (1986), "A new approach to causal inference in mortality studies with a sustained exposure periodapplication to control of the healthy worker survivor effect," *Mathematical Modelling*, 7, 1393–1512.

Robins, J. M. (1993), "Information recovery and bias adjustment in proportional hazards regression analysis of randomized trials using surrogate markers," in *Proceedings of the Biopharmaceutical Section, American Statistical Association*, American Statistical Association, vol. 24 (3).

— (1997), "Causal inference from complex longitudinal data," in *Latent variable modeling and applications to causality*, Springer, pp. 69–117.

— (2004), "Optimal structural nested models for optimal sequential decisions," in *Proceedings of the second Seattle Symposium in Biostatistics*, Springer, pp. 189–326.

Rubin, D. B. (1974), "Estimating causal effects of treatments in randomized and nonrandomized studies." *Journal of educational Psychology*, 66, 688.

Schiele, B. and Crowley, J. L. (1996), "Object recognition using multidimensional receptive field histograms," in *Computer VisionECCV'96*, Springer, pp. 610–619.

Steinwart, I. and Chirstmann, A. (2008), *Support Vector Machines*, Springer.

Steinwart, I. and Scovel, C. (2007), "Fast rates for support vector machines using Gaussian kernels," *Annals of Statistics*, 35, 575–607.

Stinchcombe, T. E. and Socinski, M. A. (2008), "Considerations for second-line therapy of non-small cell lung cancer," *The oncologist*, 13, 28–36.

Sutton, R. S. and Barto, A. G. (1998), *Reinforcement learning: An introduction*, vol. 1, Cambridge Univ Press.

Swain, M. J. and Ballard, D. H. (1992), "Indexing via color histograms," in *Active Perception and Robot Vision*, Springer, pp. 261–273.

Thall, P. F., Millikan, R. E., and Sung, H. G. (2000), "Evaluating multiple treatment courses in clinical trials," *Statistics in medicine*, 19, 1011–1028.

Thall, P. F., Sung, H. G., and Estey, E. H. (2002), "Selecting therapeutic strategies based on efficacy and death in multicourse clinical trials," *Journal of the American Statistical Association*, 97.

Thall, P. F., Wooten, L. H., Logothetis, C. J., Millikan, R. E., and Tannir, N. M. (2007), "Bayesian and frequentist two-stage treatment strategies based on sequential failure times subject to interval censoring," *Statistics in medicine*, 26, 4687–4702.

Tsitsiklis, J. N. and Van Roy, B. (1996), "Feature-based methods for large scale dynamic programming," *Machine Learning*, 22, 59–94.

Wang, L., Zhu, J., and Zou, H. (2006), "The doubly regularized support vector machine," *Statistica Sinica*, 16, 589–615.

Watkins, C. (1989), "Learning from delayed rewards." Ph.D. thesis, University of Cambridge.

Watkins, C. and Dayan, P. (1992), "Q-learning," *Machine learning*, 8, 279–292.

Weston, J., Elisseeff, A., Schölkopf, B., and Tipping, M. (2003), "Use of the zero-norm with linear models and kernel methods," *Journal of Machine Learning Research*, 3, 1439–1461.

Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T., and Vapnik, V. (2001), "Feature Selection for SVMs," *Neural Information Processing Systems 13*, 1439–1461.

Williamson, R. C. (2000), "Entropy numbers of linear function classes," in *In N. Cesa-Bianchi and S. Goldman (Eds.), Proceedings of the 13th Annual Conference on Computational Learning Theory*, Morgan Kaufman, pp. 309–319.

Yan, J., Cheng, Y., Fine, J. P., and Lai, H. J. (2010), "Uncovering symptom progression history from disease registry data with application to young cystic fibrosis patients," *Biometrics*, 66, 594–602.

Zhang, H. H. (2006), "Variable selection for support vector machines via smoothing spline ANOVA," *Statistica Sinica*, 16, 659–674.

Zhang, H. H., Ahn, J., and Lin, X. (2006), "Gene Selection Using Support Vector Machines With Nonconvex Penalty," *Bioinformatics*, 22, 88–95.

Zhang, T. and Bartlett, L. (2002), "Covering Number Bounds of Certain Regularized Linear Function Classes," *Journal of Machine Learning Research*, 2, 527–550.

Zhao, Y., Kosorok, M. R., and Zeng, D. (2009), "Reinforcement learning design for cancer clinical trials," *Statistics in Medicine*, 28.

Zhao, Y., Zeng, D., Laber, E. B., and Kosorok, M. R. (2014), "New statistical learning methods for estimating optimal dynamic treatment regimes," *Journal of the American Statistical Association*, just-accepted.

Zhao, Y., Zeng, D., Rush, A. J., and Kosorok, M. R. (2012), "Estimating individualized treatment rules using outcome weighted learning," *Journal of the American Statistical Association*, 107, 1106–1118.

Zhao, Y., Zeng, D., Socinski, M. A., and Kosorok, M. R. (2011), "Reinforcement learning strategies for clinical trials in nonsmall cell lung cancer," *Biometrics*, 67, 1422–1433.

Zhu, J., Rosset, S., Hastie, T., and Tibshirani, R. (2003), "1-norm Support Vector Machines," in *Neural Information Processing Systems*, MIT Press, p. 16.

Zou, H. and Yuan, M. (2006), "The F-∞ norm support vector machine," *Statistica Sinica*.